

Olena Yakovleva¹, Kateryna Nikolaieva²

¹ Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

² SYTOSS Ltd, Kharkiv, Ukraine

RESEARCH OF DESCRIPTOR BASED IMAGE NORMALIZATION AND COMPARATIVE ANALYSIS OF SURF, SIFT, BRISK, ORB, KAZE, AKAZE DESCRIPTORS

Abstract. The subject of research is image normalization based on key points analysis. The purpose is development of mathematical models and their software implementation for normalization of image geometric transformations based on the analysis of SIFT, SURF, ORB, BRISK, KAZE, AKAZE descriptors; the model application for comparative analysis of descriptors based on expert assessments of normalization quality, time costs and other indicators; construction and usage in experiments the own dataset with 100 real image pairs which contains scenes of five types: buildings, plane images outside, plane images inside, natural and artificial textures; making conclusions about the performance of the considered descriptors to solve the normalization problem. Such methods are applied: SIFT, SURF, ORB, BRISK, KAZE, AKAZE descriptors for describing key points, the Nearest Neighbor Distance Ratio method or symmetric method for search of corresponding pairs of key points from different images, the RANSAC method for rejecting false correspondences and obtaining a homography matrix, similarity measures, software modeling. The results obtained: experimental normalization results by SIFT, SURF, ORB, BRISK, KAZE, AKAZE descriptors for 100 real pairs of own dataset (normalized images, their overlaps, quantitative descriptor evaluation, precision and recall estimation, time costs estimation, expert quality assessment, conversion of all indicator values to an 8-point rating scale); summary diagrams and conclusions about advantages and weaknesses of the compared descriptors; recommendations about the most-suitable-algorithm selection for solving normalization problem in specific cases.

Keywords: normalization of geometric transformation; detector; descriptor; SURF; SIFT; BRISK; ORB; KAZE; AKAZE; RANSAC method; method of k-nearest neighbors; Nearest Neighbor Distance Ratio method; OpenCV library.

Introduction

Review of the current state of solving computer vision problems. Today, computer vision is experiencing a period of significant progress in problem solving, on which researchers have been working for decades. Meaningful advances have been made in detection, segmentation, classification of objects, image search by pattern, image generation, and other tasks.

For many years, most approaches of image analysis have been based on transition from an image to a feature space, which contains basic information about the whole image or its characteristics, and has a denser representation than the image. After the transition, such features are compared with the appropriate method, and then a decision is made according to the problem which is being solved. In addition to being informative and having reduced dimension, the obtained feature should be as insensitive as possible, i.e., invariant to image geometric transformations, changes in lighting, noise, local occlusion, for example, when objects are partially overlapped, or an object part extends beyond an image.

There isn't any universal way to describe images with a feature set. The choice of feature space and their processing method depends on a specific problem. The weakness of the classical feature approaches is the need for complex configuration, where it is necessary to set various parameters on the basis of heuristic information. Values of such parameters affect the final result significantly [1, 2].

In recent years, neural networks have provided a significant breakthrough in computer vision. In 2012, the AlexNet neural network took part in the annual ImageNet competition and showed the best results in

solving the object classification problem with a number of errors 15.3% against 26.2% of the runner-up [3]. Then, in 2019, the classification quality with neural networks equaled human capabilities. The combination of achievements of classical analytical feature approach with a neural network one has a great perspective.

But even now, there are a lot of tasks where the classical feature approach is necessary. Such tasks, as cartography, stitching of panoramic images etc., depend on solving the problem of normalization (compensation) of present geometric transformations.

Normalization problem and approaches to solve it. In the work, the normalization means the process of compensation of geometric transformations that distinguish one image from another. This matter has been investigated for a long time. The fundamental work [2] about normalization proposes two main approaches: tracking and parametric and also considers some methods for each of these approaches.

The tracking approach implies the gradual compensation of geometric transformations with many steps. The processed image is compared with the pattern at each step, and then it undergoes a tiny geometric transformation that compensates only a part of the whole geometric transformation, bringing the processed image closer to the pattern. As a result of all steps, the processed image will become a pattern, and parameters of a general geometric transformation will be defined. This approach is applied to tracking and targeting tasks.

The parametric approach is aimed to determine parameters of an entire geometric transformation at once. Then the found transformation is compensated, and the processed image turns into the pattern. This approach is used more widely.

In [2], for both approaches, it is offered to put an integrated method of construction of functionals which are based on the moments of a different order, but only cases with simple geometrical transformations are considered. Also, for all integrated methods, a significant problem is the background that can be partially or completely changed. In [4,5], to solve the problem of normalization in the conditions of complex geometric transformations and local occlusions, it is proposed to use the method of one-dimensional normalizations and decomposition of complex groups of transformations into compositions of simple ones. However, such methods solve the problem only partially. This article is devoted to analyzing the normalization on the basis of the descriptors of image key points.

Construction of image features based on descriptors. In the classical approach, a solution of a big amount of tasks is based on the key points definition and description of their neighborhoods by a feature vector with further processing of obtained vectors. So, there is a transition from an image to a space of key points feature vectors.

An algorithm that gets key points is a detector, and an algorithm that gets a description of the found points is called a descriptor. Also, a descriptor means a feature vector of a key point.

Over a long period of existence of computer vision tasks, a significant number of algorithms have been developed to detect and describe key points, which differ in varying degrees of invariance to geometric transformations, changes in lighting, angles of view, and time costs values. The implementation of most of these algorithms can be seen in popular software libraries. For instance, the open library OpenCV contains SURF, SIFT, ORB, BRISK, KAZE, AKAZE, LATCH, VGG, LUCID, DAISY, FREAK and other descriptors.

Statement of the research task. This work is dedicated to:

- research of the parametric normalization approach, where key points and their descriptors are used to find out the normalization parameters;
- comparing the quality and time costs of the normalization process based on different descriptors. For comparison were selected the full-cycle descriptors, such as SURF128, SURF64, SIFT, BRISK, ORB, ORB1000, KAZE, AKAZE.

In this work, it was necessary:

- to consider each step of normalization in detail;
- to develop a mathematical and software model for further experiments on the photo image pairs normalization with the different descriptors;
- to create the own dataset for experiments;
- to draw conclusions in the comparative aspect about the quality and time costs of normalization basing on the considered descriptors.

Main material of the research

1. Normalization of geometrical transformations based on the descriptors. The normalization method based on descriptors uses the basic property of projective transformation

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}$$

about the possibility of obtaining parameters from coordinates of 4 points before and after the transformation [6, 7]:

$$\begin{cases} h_{13} = x_{A'}; \\ h_{31} = \frac{d_1(y_{C'} - y_{B'}) - d_2(x_{C'} - x_{B'})}{AB(d_3 - d_4)}; \\ h_{11} = \frac{x_{D'}(h_{31}AB + 1) - h_{13}}{AB}; \\ h_{21} = \frac{y_{D'}(h_{31}AB + 1) - h_{23}}{AB}; \\ h_{23} = y_{A'}; \\ h_{32} = \frac{d_1(y_{C'} - y_{D'}) - d_2(x_{C'} - x_{D'})}{AD(d_4 - d_3)}; \\ h_{12} = \frac{x_{B'}(h_{31}AD + 1) - h_{13}}{AD}; \\ h_{22} = \frac{y_{B'}(h_{32}AD + 1) - h_{23}}{AD}; \end{cases}$$

where $h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}$ – parameters of projective transformation; $A(x_A, y_A)$, $B(x_B, y_B)$, $C(x_C, y_C)$, $D(x_D, y_D)$ and $A'(x_{A'}, y_{A'})$, $B'(x_{B'}, y_{B'})$, $C'(x_{C'}, y_{C'})$, $D'(x_{D'}, y_{D'})$ – 4 points before and after the transformation on B_1 and B_2 images respectively, AB, AD – segment lengths and

$$\begin{aligned} d_1 &= x_{D'} - x_{C'} + x_{B'} - h_{13}, \\ d_2 &= y_{D'} - y_{C'} + y_{B'} - h_{23}, \\ d_3 &= (x_{C'} - x_{D'})(y_{C'} - y_{B'}), \\ d_4 &= (y_{C'} - y_{D'})(x_{C'} - x_{B'}). \end{aligned}$$

But in practice, the corresponding points on B_1 and B_2 images are unknown. Descriptors that detect and describe the key points can be used to solve the problem of establishing the corresponding points. Key points will be found with some inaccuracy because detectors and descriptors are sensitive to significant geometric and lighting transformations. The search of corresponding points that is defined based on the descriptor similarity will also make false pairs.

Therefore, in practice, it is desirable to use more than 4 pairs of found corresponding points to determine the geometric transformation parameters more accurately.

The normalization algorithm used in the work consists of the following steps:

- 1) key points search and their description with feature vectors, i.e., the descriptors, for B_1 and B_2 images;
- 2) definition of the correspondence between the key points on B_1 and B_2 images;

3) determination of geometric transformation parameters, which distinguishes B_2 image from B_1 image;

4) B_2 image normalization to B_1 image (direct normalization), or B_1 image normalization to B_2 image (inverse normalization).

Further, let's consider each step in detail.

Step1. To define key points and their description, the work considers full-cycle algorithms (detector-descriptor algorithms), which perform both key points detection and description. Based on the multi-source analysis, the detector-descriptor algorithms SURF128, SURF64, SIFT, BRISK, ORB, ORB (1000), KAZE, AKAZE were chosen as the most perspective and interesting for normalization. Table 1 shows the brief information about these algorithms. The principles of constructing descriptors according to these algorithms can be found at the links [8–13].

However, a large number of existing algorithms and the lack or inconsistency of information about their comparison, recommendations for their use in different conditions make it difficult to understand strengths and weaknesses of certain descriptors and to choose the best for solving own task. Thus, there is a need for further descriptor research in the comparative aspect and obtainment of sound recommendations for their usage.

Table 1 – Brief information about the considered full-cycle descriptors (detector-descriptor algorithms)

Name of full cycle descriptors, year, source, type of license	Detector: type; Descriptor: type, size
SIFT (Scale Invariant Feature Transform), 2004, [8], non-commercial use	detector: blob; descriptor: numeric, 128
SURF (Speeded Up Robust Features) 2008, [9], non-commercial use	detector: blob; descriptor: numeric, 128 or 64
ORB (Oriented FAST and Rotated BRIEF)/ORB1000/ 2011, [10], free use	detector: corner; descriptor: binary, 32
BRISK (Binary Robust Invariant Scalable Keypoints) 2011, [11], free use	detector: corner; descriptor: binary, 64
KAZE 2012, [12], free use	detector: blob; descriptor: numeric, 128
A-KAZE (Accelerated –KAZE) 2013, [13], free use	detector: blob; descriptor: binary, 61

Mentioned below, the term “descriptor” is employed addressing detector-descriptor algorithms, excluding some cases where it is necessary to discuss the detector part of the full algorithm. The term “descriptor” is also used when it refers directly to a key point feature vector that was received as the result of applying the algorithm.

So, one of the detector-descriptor algorithms constructs the sets

$$D_1 = \{(M_1(x, y), d_1)\} \text{ and } D_2 = \{(M_2(p, q), d_2)\}$$

for images B_1 and B_2 respectively, where $M_1(x, y)$ – a key point, d_1 – a descriptor (vector with features

c_{1v}) for B_1 image; $M_2(p, q)$ – a key point, d_2 – a descriptor (vector with features c_{2v}) for B_2 image; $v = 1 \dots V$, V – the dimension of the feature vector, which is determined by the certain descriptor method.

Step 2. To establish the matches between key points, one should compare their descriptor vectors. As the result of matching the set of corresponding pairs

$$T = \{(M_{1i}(x, y), M_{2j}(p, q))\}, (|T| \leq \min(|D_1|, |D_2|))$$

is formed. To search for corresponding pairs, the Nearest Neighbor Distance Ratio (NNDR) method (modification of k-nearest neighbors, k=2) was used, and the symmetric method was covered in order to analyze some estimates of normalization.

The NNDR method consists of finding for each point $M_{1i}(x, y)$ and its descriptor d_{1i} on the image B_1 two most similar descriptors d_{2k} and d_{2t} on the image B_2 . To measure the similarity of numerical descriptors was applied Manhattan distance (L1 norm)

$$\rho(d_{1i}, d_{2j}) = \sum_{v=1}^V |c_{1i_v} - c_{2j_v}|,$$

which becomes Hamming distance for binary descriptors.

The point M_{2k} with the descriptor d_{2k} of B_2 image was considered the match the point M_{1i} with the descriptor d_{1i} of B_1 image, if the inequality

$$\frac{\rho(d_{1i}, d_{2k})}{\rho(d_{1i}, d_{2t})} < \sigma,$$

is satisfied, where $\rho(d_{1i}, d_{2k}) < \rho(d_{1i}, d_{2t})$; σ – threshold (σ was equal to 0.75 in the experiments), and a pair of points (M_{1i}, M_{2j}) was put into the set T .

After processing all key points of B_1 image, the same actions were performed for B_2 image key points which weren't determined as the corresponding pair for the image B_1 .

The found corresponding pairs for B_2 image were added to the set T .

In this approach, the result set T depends on the image which is processed first. Namely, if at first B_2 image key points are treated and at second B_1 image key points are done, the set T will be different.

In practice, the symmetric approach for finding corresponding pairs has also been researched, where the set T consists only of such pairs of points (M_{1i}, M_{2j}) for which the similarity measure is the least

$$\begin{aligned} \rho(d_{1i}, d_{2j}) &= \\ &= \min_{a=1, \dots, |D_2|} \rho(d_{1i}, d_{2a}) = \min_{b=1, \dots, |D_1|} \rho(d_{1b}, d_{2j}). \end{aligned}$$

The symmetrical approach does not depend on the order of B_1, B_2 image processing.

The results of the research have shown that the use of the NNDR method or the symmetric approach at the step of matching gives a similar accuracy of the final normalization, but the symmetric method has higher time costs, almost twice (Fig. 9). The reason for this is the fact that the symmetric approach finds much more matches, but among the found matches, the false match (outlier) share is significantly larger than in the NNDR method (Fig. 1,2). This is also critical in the conditions of a small number of obtained key points. So, the main attention was paid to the NNDR method, and the symmetric method was considered only in some cases.

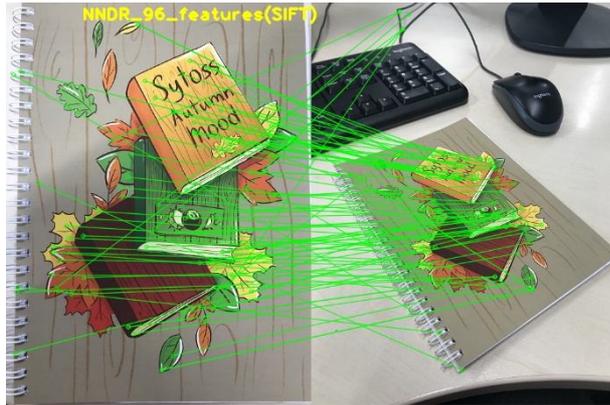


Fig. 1. The corresponding pairs of key points were obtained with the NNDR method based on the SIFT descriptor (96 matches)



Fig. 2. The corresponding pairs of key points were got with the symmetric method based on the SIFT descriptor (217 matches)

Step 3. If there is a set T of corresponding pairs of points, the geometric transformation parameters can be got on the basis of the least squares method. But the set T , besides correct matches (inliers), may contain false correspondences (outliers), that is confirmed by practical results (Fig. 1,2).

Such outliers would significantly affect a geometric transformation model made with the least squares method, and this model would not be suitable for normalization.

Therefore, the RANSAC method was applied in this work. It is designed to look for the best homography matrix without taking outliers into consideration. To be more precise, the accelerated

implementation of this method supported by the OpenCV library, which was proposed in [14], was used.

Thereby, the applied in this paper algorithm for search of the geometrical transformation model consists of two stages:

Stage I. Search for the best initial model without considering outliers with the RANSAC method using a geometric test. The RANSAC input is supplied with the set T of corresponding pairs of points for B_1 and B_2 images (constructed with the NNDR or the symmetric method); the number of iterations N ; the threshold δ for estimation of match error (discarding outliers).

Repeat N times (in this work $N = 2000$), on each k -th iteration:

a) select randomly 4 corresponding pairs of points (M_{1i}, M_{2j}) from the set T .

b) check the selected subset for compliance with the geometric test [14]. For this, every 3 pairs of points from the subset formed in a) (every 3 pairs from 4 selected ones) are examined for the coincidence of the traversal sequence on B_1 and B_2 respectively.

The time for implementation of the entire RANSAC algorithm can be significantly reduced with the help of b), because if the points selected in a) do not fit, then c), d), e) are omitted, and we return to a); otherwise we move to c).

c) calculate the homography matrix H_k for the selected 4 corresponding pairs of points.

d) define quality U_k of the model H_k , namely, how accurately the matrix H_k transforms all corresponding pairs of points from the set T . Model quality U_k is calculated as the number of cases for which the match error e is less than the threshold δ

$$e(M'_{1i}, M_{2j}) < \delta, \quad (1)$$

where $M'_{1i} = H_k M_{1i}$ – a reflection of point M_{1i} of the image B_1 obtained with the homography H_k ; $(M_{1i}, M_{2j}) \in T$ – the corresponding pair; δ – threshold (equal 3 in the experiments); $e(\cdot)$ – the match error. For example, the error can be computed as Euclidean distance between a key point on the image B_2 and reflection of the corresponding key point on the image B_1 or as the total error of direct transformation H_k (direct error) and inverse one H_k^{-1} (transfer error). Pairs of points corresponding, for which (1) is true, are considered to be correct matches (inliers), others – false matches (outliers).

e) find the model H_k that has the best quality U_k ($U_k \geq U_{max}$), store it as $H^* = H_k$ and construct the set T^* by removing from the set T outliers.

At the end of the cycle, we obtain the best model H^* and the inlier set T^* .

Stage II. Refinement of the initial model for the inlier set T^ with the method of least squares.* In this

stage, it is determined the homography matrix H^{best} , for which the sum of deviations between the key points from the set T^* on the image B_2 and the reflection of the matching key points on the image B_1 is minimum:

$$\sum_{T^*} \left[\left(x_{2j} - \frac{h_{11}x_{1i} + h_{12}y_{1i} + h_{13}}{h_{31}x_{1i} + h_{32}y_{1i} + h_{33}} \right)^2 + \left(y_{2j} - \frac{h_{21}x_{1i} + h_{22}y_{1i} + h_{23}}{h_{31}x_{1i} + h_{32}y_{1i} + h_{33}} \right)^2 \right],$$

where $h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}$ – parameters of the model H^{best} , x_{1i}, y_{1i} – the coordinate of point $M_{1i} \in B_1$, x_{2j}, y_{2j} – the coordinate of point $M_{2j} \in B_2$, $(M_{1i}, M_{2j}) \in T^*$.

Step 4. The direct normalization is based on the transformation of B_1 with the matrix H^{best} . The result of normalization is the normalized image B_1^H . Relatively, the inverse normalization is the transformation of B_2 with the inverse matrix $(H^{best})^{-1}$.

2. Software implementation of the normalization method based on SIFT, SURF128, SURF64, BRISK, ORB, ORB (1000), KAZE, AKAZE descriptors. All experiments were performed using Java language and OpenCV library. OpenCV methods, shown in Table 2, were used to obtain descriptors.

Table 2 – OpenCV methods for obtaining descriptors

Algorithm name	Element type of vector	OpenCV object
SIFT	CV_32F	SIFT sift = SIFT.create();
SURF(128D)	CV_32F	SURF surf128 = SURF.create(); surf128.setExtended(true);
SURF (64D)	CV_32F	SURF surf64 = SURF.create(); surf128.setExtended(false);
ORB	CV_8U	ORB orb = ORB.create(); orb.setMaxFeatures(100000);
ORB (1000)	CV_8U	ORB orb1000 = ORB.create(); orb1000.setMaxFeatures(1000);
KAZE	CV_32F	KAZE kaze = KAZE.create(); kaze.setNOctaveLayers(3); kaze.setExtended(true);
AKAZE	CV_8U	AKAZE akaze = AKAZE.create(); akaze.setNOctaveLayers(3);
BRISK	CV_8U	BRISK brisk = BRISK.create();

OpenCV class BFMatcher(int normType, bool crossCheck) was used to find matches, where

crossCheck = false – for the k-nearest neighbor method, true – for the symmetric approach; normType = NORM_L1 (Manhattan distance) – for numeric descriptors, NORM_HAMMING (Hamming distance) – for binary ones. BFMatcher class has the bfMatcher.knnMatch() method for implementing k-nearest neighbors and the bfMatcher.match() method for the symmetric approach. The input parameters are descriptor sets of the images B_1 and B_2 and KNN_MATCH_COUNT – parameter k, k=2 (for the k-nearest neighbors). After using the k-nearest neighbor method, filterMatchesByNNDR() method should be performed for implementing the NNDR method. At the output, we get matches – the set of corresponding point pairs.

The geometric transformation parameters were searched using findHomography() function, which implements both stages considered in the paragraph 1 (initial model search, outlier rejection and model refinement for inliers). The input findHomography() parameters are as follows:

- the coordinate set of the corresponding points of images B_1 and B_2 (obtained by executing bfMatcher.knnMatch() or bfMatcher.match());
- the method for finding homography (this parameter was equal CV_RANSAC() for the application of the RANSAC method in the work);
- the threshold for checking inequality (1) which was equal 3;
- an empty set to mark the outliers and inliers by 0 and 1, respectively.

The output is a homography matrix and the completed set of outliers and inliers.

3. Dataset description for research of descriptor-based normalization.

The SYTOSS_NURE_pngPairs100 dataset was created to research the normalization approach based on the analysis of key points. It has 100 pairs of self-made photos with a Sony Alpha a6000 camera, converted to PNG format (lossless compression) and reduced to a size of 600×400 or 400×600 pixels (Fig. 3). The original raw images in ARW format can be found here [15]. The dataset consists of 5 sets of image pairs, and every set contains scenes of the certain type (20 images in each set): Building – buildings, city; Picture_outside – plane images outside (graffiti, posters and other plane images found at the streets); Picture_inside – plane images inside (for example, interior images, pictures, books); Texture_artificial – artificial textures; Texture_nature – natural textures. In each pair, the images differ by geometric transformations of variable complexity (displacement, scale, rotation, change of viewpoint).

The purpose of experiments with the SYTOSS_NURE_pngPairs100 dataset is to analyze in the comparative aspect the normalization results based on SIFT, SURF, ORB, BRISK, KAZE, AKAZE descriptors for image pairs containing scenes of different types taken with the same equipment, PNG format (without compression); also, to make a general conclusion about the most suitable descriptors, using the normalization of a large number of real image pairs.

To test the descriptor-based normalization in the most complex lighting transformation, we used the small set Day_Night_pngPairs3, which consists of 3 image pairs with the SYTOSS_NURE_pngPairs100 dataset properties (equipment, size, format, geomantic transformation), but in each pair, one image is shot at day and the other at night (Fig.4).

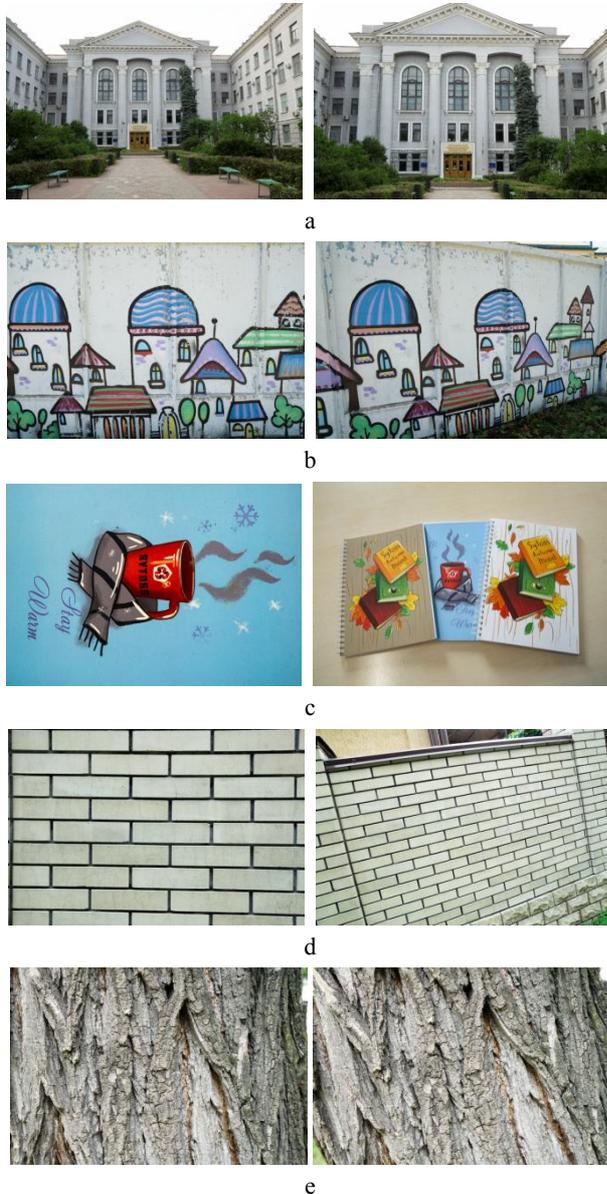


Fig. 3. Image pairs examples for each set from the SYTOSS_NURE_pngPairs100 dataset:
 a – Building; b – Picture_outside; c – Picture_inside;
 d – Texture_artificial; e – Texture_nature

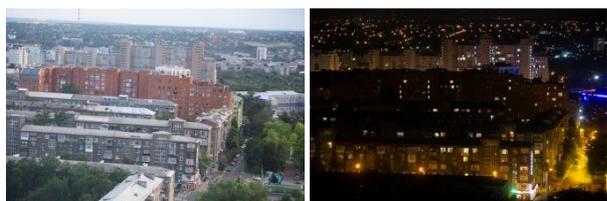


Fig. 4. Image pair example from the Day_Night_pngPairs3

4. The purpose and content of experiments. The purpose of the research is to identify the most

appropriate descriptors for solving the normalization problem. To achieve this goal, a comparative evaluation of quantitative indicators for SIFT, SURF128, SURF64, BRISK, ORB, ORB (1000), KAZE, AKAZE descriptors was performed at each step of normalization (key points search, matches, outlier rejection and obtaining normalization parameters), and what was even more important, it was compared the normalization quality and time costs in general.

To estimate the normalization results based on the considered descriptors, the indicators shown below were calculated for each pair.

4.1. Quantitative evaluation of descriptors, precision and recall estimation (with taking found overlap):

NP – a number of found key points; NP1, NP2 – a number of found key points for image1 and image2, respectively, NPO1, NPO2 – a number of found key points on the overlap for image1 and image2, respectively;

NM – a number of found matches, i.e., a number of obtained corresponding pairs of key points (NM=NI+NO); NMO – a number of matches located on the overlap;

NI – a number of inliers found with the RANSAC method, NO – a number of outliers discarded with the RANSAC method;

Precision = NI/ NM – the accuracy of finding correct matches. Precision ratio defines a part of inliers to all found matches, including outliers. This illustrates the ability of the algorithm to identify points correctly and provide the most similar description for corresponding points and vice versa dissimilar description for inappropriate points.

The conclusion depends on the method used to find the corresponding pairs.

RecallO1 = NI/ NPO1 – the completeness of inlier retrieval relative to the number of all key points on the overlap for the pattern (image1). RecallO1 ratio illustrates the usefulness of the found key points for normalization. It is also directly related to time costs because it gives the possibility to estimate how much time was wasted on detecting, describing and comparing points, which were then discarded and did not participate in the construction of the normalization model.

4.2. Estimation of time costs:

DesT – the descriptor construction time (time to detect and describe key points on an image), DesT1, DesT2 – the construction time of descriptors for image1 and image2, respectively;

MatchT - the retrieval time of matches for an image pair;

InlierT – the inlier retrieval time found with the RANSAC method for an image pair;

avgDesT – the average time for one descriptor construction;

avgMatchT – the average time for one match retrieval;

avgInlierT – the average time for one inlier retrieval with the RANSAC method;

TotalNormT – total normalization time for an image pair.

4.3. Expert quality assessment. In this work, the expert rate was used for the assessment of normalization quality. The expert rate (ER) is the expert quality assessment of normalization accuracy with a 5-point scale: “0” – normalization failed, “1” – insufficient, “2” – satisfactory, “3” – good, “4” – excellent.

Seven independent experts participated in the 5-point scale assessment of the normalization quality of each image pair for every descriptor. The experts assessed normalization quality as the overlap accuracy of normalized and original pair images. Experts had to evaluate the normalization accuracy in a comparative aspect (relatively to the results obtained by other descriptors for the same pair). If the expert believed that the normalization quality got by different descriptors for a certain pair is comparable, one put the same rate to these descriptors. In the case, when all descriptors coped equally, the expert gave the same rate to all descriptors.

The median of the expert rates of 7 experts was taken as the final assessment quality of the experiment.

The expert rate is a relative indicator given to the normalization results respectively to other descriptors. Thus, it allows ranking the descriptors from worst to best in one experiment (for a certain pair).

Value “-1” was assigned if the normalization could not be performed due to the lack of a sufficient number of matches to determine the homography matrix.

In this research, expert assessments were used to estimate the normalization quality because only a human can assess the quality of the normalized and original image overlap most accurately. However, this approach is time-consuming and subjective. In the future, it is planned to conduct a research to define some quantitative rates of normalization quality, which are calculated after normalizing automatically and correlate with the expert assessment maximally.

4.4. Averaging the values of indicators and the rates. The above-mentioned rates and indicators were calculated for 100 real image pairs from the SYTOSS_NURE_pngPairs100 dataset. Then for each of 5 sets (Building, Picture_outside, Picture_inside, Texture_artificial, Texture_nature), as well as for the whole dataset, the mean values were computed:

- mean values of absolute indicators NP, NM, NI;
- mean values of relative indicators Precision, RecallO1;
- mean values of time costs indicators DesT; MatchT; InlierT; AvgDesT; AvgMatchT; AvgInlierT; TotalNormT;
- the quantity of the same values of the expert rate ER for each value (“0”, “1”, “2”, “3”, “4”), as well as the amount of cases where the normalization did not occur due to the lack of inliers (values “-1”);
- mean values of expert rates ER for each descriptor (arithmetic mean MeanER and median).

4.5. Converting all indicator values from 4.4 to an 8-point rating scale. To summarize and compare the descriptors for different indicators easier, all assessment values were converted to an 8-point scale (point 8 is the highest mark). One of the values from 1 to 8, depending on which interval it got into, was assigned to each value of the indicator. The interval was calculated using the

following formula:

$$(\min + i * \text{step}; \min + (i+1) * \text{step})$$

where $\text{step} = (\max - \min) / 8$, \max , \min – maximum and minimum values of indicator respectively, $i = 0, \dots, 7$. If a larger value was considered the best for an indicator, then the highest score 8 was assigned to the values from the last interval ($\min + 7 * \text{step}$, \max]. Vice versa, if a lower value was regarded as the best, then the highest score 8 was assigned to the values from the first interval [\min , $\min + \text{step}$].

5. Normalization research results based on the descriptors SIFT, SURF128, SURF64, BRISK, ORB, ORB(1000), KAZE, AKAZE. The limited volume of this paper makes it possible to present only generalized results for the whole dataset. The details of normalization results for each pair, as well as, many summary tables and diagrams for each of the sets and the entire dataset, can be found here [15]. The normalization research results averaged for the whole dataset are shown in paragraph 5.1–5.4. The main received regularities for each type of scenes are illustrated in 5.5.

5.1. Comparison of descriptors by the mean number of key points, matches (with the NNDR method) and inliers. The experiments showed that a significant excess of the mean NP is characteristic for the ORB algorithm (from 2.6 times for the BRISK algorithm to 11 ones for the AKAZE algorithm) (Fig. 5). Sorting out the mean values of NP, one can see that, according to the number of found points, the algorithms are placed in the following order (from larger NP to smaller one): ORB >> BRISK >> SURF64, SURF128, SIFT >> ORB1000 > KAZE, AKAZE.

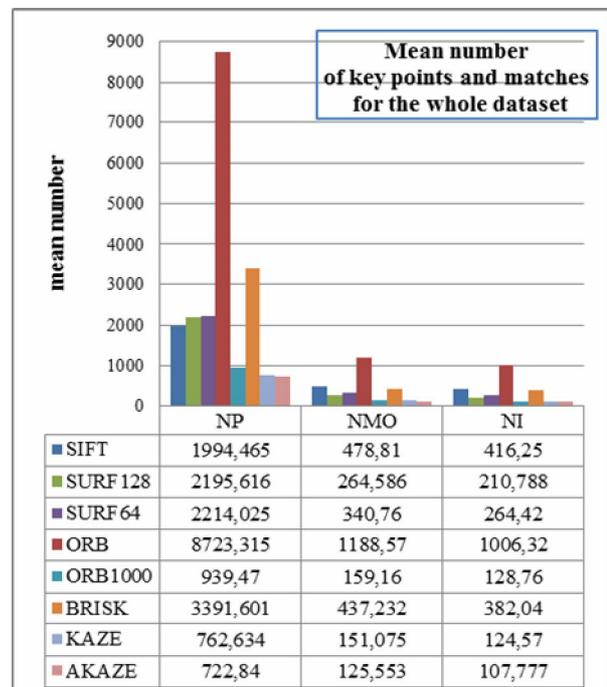


Fig. 5. Mean number of key points and matches found at the first three stages of normalization for SYTOSS_NURE_pngPairs100, i.e., the mean of such rates: a number of found key points (NP); a number of matches found with the NNDR method and located on the overlap (NMO); a number of inliers found with the RANSAC method (NI)

The largest and least mean values of NMO, NI are the same as for NP: the largest is for the descriptor ORB, the least – for ORB1000, KAZE, AKAZE that can be seen in Fig.5. This figure also shows the numerical values of mean NP, NMO, NI. Changes of NMO, NI values for different descriptors have the same regularity. Descriptors can be arranged by NI (from larger to smaller) as follows: ORB >> SIFT > BRISK > > SURF64 > SURF128 > ORB1000, KAZE > AKAZE, where the SIFT algorithm is seen moved from 5th position (by found NP) to 2nd position, overtaking BRISK, SURF64, SURF128, i.e., it has the higher percentage of key points which will be used for normalization.

The descriptor evaluation by NP, NMO, NI was carried out under the assumption that the greater the number of NP, NMO, NI, the more accurately constructed the normalization model and, therefore, the higher the normalization quality, however, too much quantity of such ones requires a lot of time that will be presented below.

5.2. Descriptor comparison by Precision and Recall ratios. The descriptor evaluation by Precision and Recall was performed supposing that:

- the higher value of Precision, the more alike description of really similar objects (points) we get with this algorithm and the more unlike description for dissimilar ones. In addition, with high Precision, there is less unnecessary work when it takes time to find matches, some of which are outliers and will be rejected with the RANSAC algorithm;

- the higher RecallO1, the less time wasting for getting key points at the first step. RecallO1 shows the percentage of key points found at the 1st step, which will be used to define a homography matrix at the 3rd step, i.e., allows estimating a share of found points actually used for normalization.

The results of Precision and RecallO1 research for the whole dataset are presented in Fig. 6.

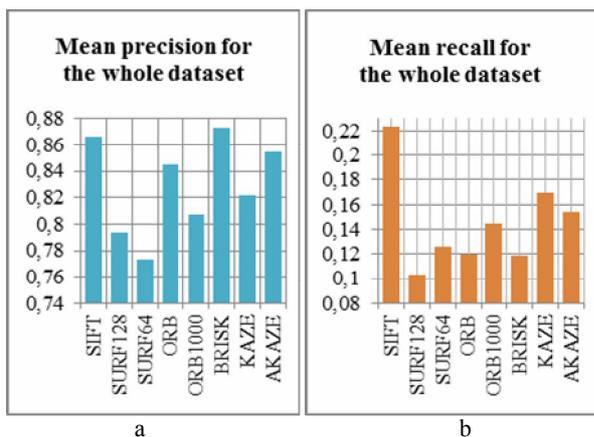


Fig. 6. Mean precision and recall for the whole dataset SYTOSS_NURE_pngPairs100: a – mean precision (Precision); b – mean recall (RecallO1)

The mean values of Precision for all descriptors vary in the range from 0.77 to 0.87, i.e., at the average for the different descriptors, the RANSAC method rejects from 23 to 13 percent of NO, which were

incorrectly defined as matches by the NNDR method (Fig. 6).

The highest (best) Precision was shown by the descriptors BRISK, SIFT, the worst one – by SURF64. In general, by Precision value, the descriptors can be sorted from the best to the worst as follows: BRISK, SIFT > AKAZE > ORB > KAZE > ORB1000 > SURF128 > SURF64.

The mean values of RecallO1 throughout the dataset show a variance in the range from 0.1 (SURF128) to 0.22 (SIFT) (Fig.6). The descriptor ORB, which has a significant excess in the number of key points comparing with the other descriptors (Fig.5), has mean RecallO1 equal 0.12, i.e., only 12% of the points found are used to determine a normalization matrix. By RecallO1, the descriptors can be ordered (from the best to the worst) as follows: SIFT >> KAZE > AKAZE > ORB1000 > SURF64 > ORB, BRISK > SURF128.

5.3. Time costs comparison. Let's consider time costs searching for a descriptor, a match (method NNDR) and an inlier (Fig. 7). According to Fig. 7, ORB shows significantly less time to calculate one descriptor (AvgDesT), then ORB (1000) and BRISK come, and after them AKAZE. SIFT, SURF64, SURF128 descriptors appear next with comparable time. The KAZE descriptor illustrates significantly more time. The average one-match search time (AvgMatchT) with the NNDR method is essentially more consuming for ORB. AKAZE and ORB(1000) has the least AvgMatchT.

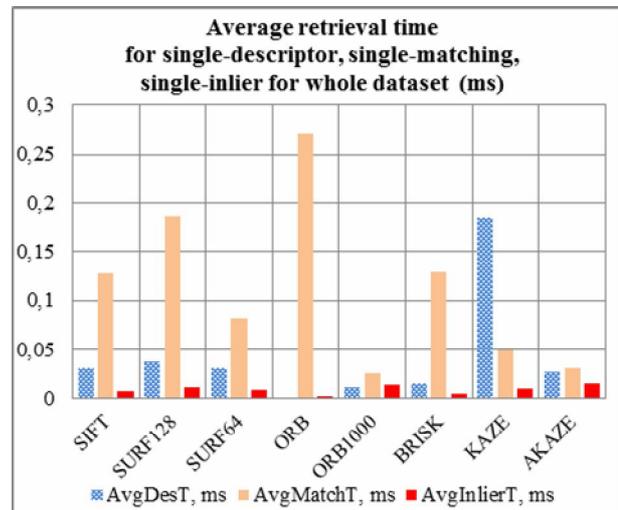


Fig. 7. Average retrieval time for a descriptor, a matching and an inlier for the whole dataset

The average retrieval time for one inlier (AvgInlierT) is longer for ORB1000 and AKAZE algorithms. For all algorithms, the values of AvgInlierT are significantly less than the values of AvgDesT or AvgMatchT. However, if we compare the values of AvgDesT and AvgMatchT for different descriptors, we do not see a single pattern. For KAZE, the time of AvgDesT is 3 times longer than AvgMatchT, and for other algorithms, AvgDesT is essentially less than AvgMatchT, but for ORB, AvgDesT is significantly less, more than 14 times.

However, the most useful for future conclusions in a comparative aspect is the total time for normalizing one image pair averaged for the whole dataset (TotalNormT). Fig.8 presents the time spent on the definition of all descriptors on the first and second images (DesT1, DesT2); all matches MatchT found with the NNDR method; all inliers InlierT for one image pair (all-time evaluations were averaged for the whole dataset).

The sum of the mean values of DesT1, DesT2, MatchT, InlierT is the mean total time (TotalNormT) of the normalization of one image pair for the dataset SYTOSS_NURE_pngPairs100 (Fig.8). Clarifying the fact that TotalNormT was estimated for a certain dataset is important because TotalNormT directly depends on the size of the images, the share of total scenes in pairs (overlap), and the type of scenes.

Below, we can see the same TotalNormT diagram for the case of using the symmetric method at the 2nd step of normalization (match search). This figure shows that the mean time for normalization almost doubles, as it was above mentioned in paragraph 2.

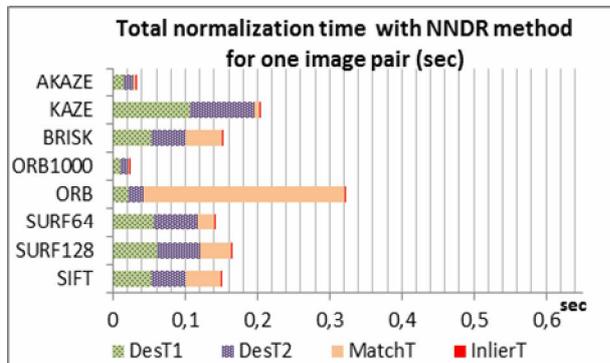


Fig. 8. Total normalization time costs of one image pair with using the NNDR method for a match search (averaged for the whole dataset)

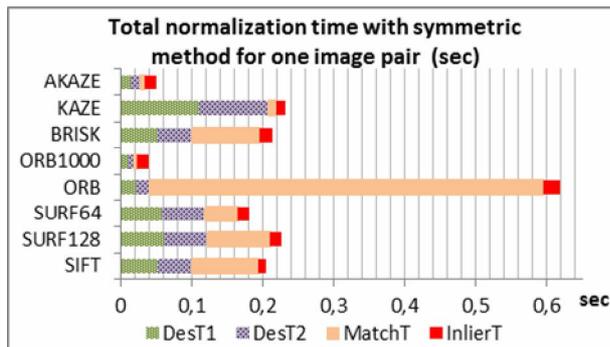


Fig. 9. Total normalization time costs of one image pair with using the symmetric method for a match search (averaged for the whole dataset)

Summing up the time costs, the descriptors by the total normalization time TotalNormT can be ranked as follows (from faster to slower):

$$ORB1000 < AKAZE \ll BRISK, SURF64 < SIFT < SURF128 \ll KAZE \ll ORB.$$

5.4. The normalization quality comparison based on expert assessments. All 800 experiments (for each pair of images 8 descriptors, 100 pairs) were

evaluated by experts (“0” –“4”). If normalization could not occur due to insufficient matches, the experiment was given the value “-1” [15]. The expert rate revealed that 68% of experiments for this dataset scored “4” and “3” (good and excellent), 7% were unsuccessful, scoring “-1” and “0” (Fig.10).

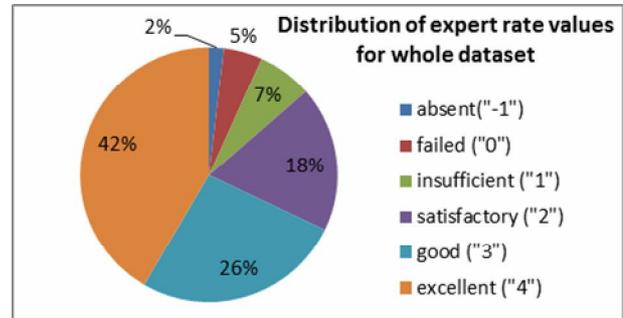


Fig. 10. Distribution of expert rate values

Fig. 11 and Table 3 illustrate how these rates were distributed among the descriptors, where it can be seen that SIFT is a clear leader. This descriptor has the biggest number of the highest rate “4”, the least number of the rate “0”, the rate “-1” is absent at all.

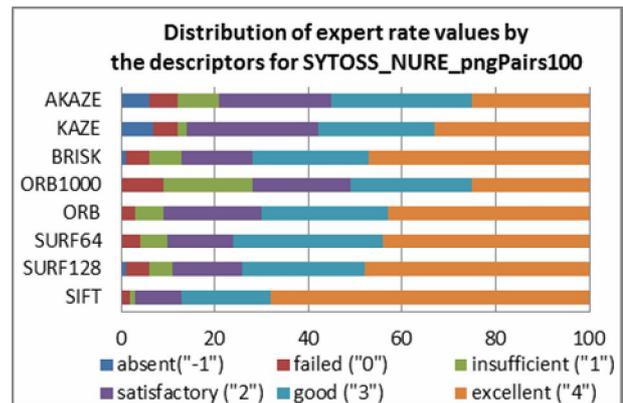


Fig. 11. Distribution of expert rate values by the descriptors for the SYTOSS_NURE_pngPairs100 dataset (for each descriptor, it is calculated the quantity of the same values of the expert rate (ER) for each value (“0”–“4”), as well as the quantity of cases where the normalization did not occur (“-1”)

Table 3 – Quantity of expert rates with the same values and mean expert rates for each descriptor (800 pairs)

Descriptor	Quantity of the same ER values						Mean ER
	-1	0	1	2	3	4	
SIFT	0	2	1	10	19	68	3,5
SURF128	1	5	5	15	26	48	3,04
SURF64	0	4	6	14	32	44	3,06
ORB	0	3	6	21	27	43	3,01
ORB1000	0	9	19	21	26	25	2,39
BRISK	1	5	7	15	25	47	2,99
KAZE	7	5	2	28	25	33	2,57
AKAZE	6	6	9	24	30	25	2,41
Total quantity	15	39	55	148	210	333	

Fig. 12, a and Table 3 show the arithmetic mean (MeanER) of expert rate values for each descriptor, where one can notice that the highest mean value 3.5 belongs to SIFT, the descriptors SURF64, SURF128, ORB, BRISK follow. ORB1000 and AKAZE have the lowest mean ones.

For SIFT only, the median of the expert rates has a value 4. If the median is calculated without the negative points “-1” and “0”, then the value 4 has SURF128 [15].

Based on the mean expert assessment and its median, the descriptors can be arranged in the following order (from a higher expert rate to lower):

SIFT >> SURF128, SURF64, ORB, BRISK >
> KAZE > AKAZE > ORB1000.

Thus, according to the experts, the SIFT descriptor has a definite advantage in the normalization quality, but it has the middle position by time consuming (Fig. 8).

This paper didn't have such a purpose to research descriptors under significant lighting changes, but the experiments with 3 pairs of Day_Night_pngPairs3 dataset have shown that SIFT presents the best result (all pairs have an expert rate “4”), and KAZE has the lowest mean expert rate (Fig. 12, b, 13, 14).

However, this issue requires additional research for more images. The results of the experiments can be found here [15].



Fig. 14. Result of normalization for the image pair from Fig. 4 (Day_Night_pngPairs3)

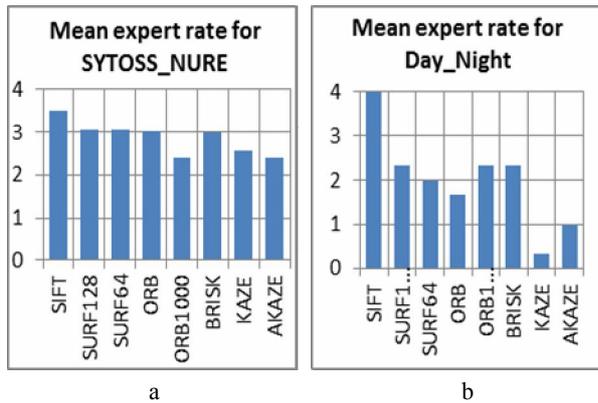


Fig. 12. Arithmetic mean of expert rate values (AvgER): a – mean expert rate for SYTOSS_NURE_pngPairs100; b – mean expert rate for Day_Night_pngPairs3

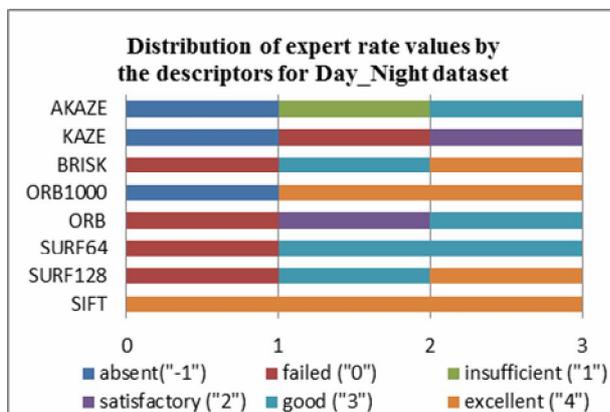


Fig. 13. Distribution of expert rate values by the descriptors for Day_Night_pngPairs3

5.5. Analysis of normalization results for scenes of different types. The paper considers how the descriptors process images of different sceneries (Building, Picture_outside, Picture_inside, Texture_artificial, Texture_nature).

As the limited volume of the printed version does not allow to illustrate all the obtained regularities, in this article, only some of them appear below:

- all algorithms are characterized by a significant increase in number of found key points NP for natural textures. This increase is particularly typical for ORB;
- ORB, BRISK descriptors have a fairly consistently high mean Precision for all sets (from 0.84 to 0.87 for ORB, from 0.85 to 0.89 for BRISK). For other algorithms, the mean Precision varies significantly from set to set. For example, the SIFT algorithm shows the highest Precision value of 0.93 for artificial textures. In general, the scatter of mean Precision for different sets is 0.09;
- except SIFT, all descriptors showed the worst RecallO1 for artificial textures, SIFT has the highest RecallO1 for all sets;
- the time indicator AvgDesT for detection and description of one key point for the descriptors ORB, BRISK and ORB100 is significantly less than for the others and does not depend on the type of scene. For the other algorithms, AvgDesT values differ depending on scene types.
- the ORB algorithm proved to be very slow working with natural textures, that affected the estimation of time cost TotalNormT with this algorithm for the whole dataset (Fig.8,9);
- the lowest mean expert rate was defined for textures, at that natural texture is processed worse with SIFT, SURF128, SURF64, ORB, ORB1000, BRISK descriptors, and artificial texture is worse with KAZE, AKAZE;
- there are 13 cases out of 15 ones for artificial texture when normalization was not possible (ER = "-1"), where 11 cases occurred under KAZE and AKAZE algorithms. The natural texture had the lowest quantity of the rate “4”. SIFT descriptor shows the highest expert rates for each set.

All results of experiments for different scene type sets can be found in [15]. In general, we conclude that

normalization with the research method gives the worst results for natural and artificial textures.

This can be explained by the fact that natural textures often have distortions partly beyond projective transformations, and the artificial textures have more quantity of similar neighborhoods for not corresponding key points. However, to confirm the identified regularities, it is necessary to increase the number of image pairs from 20 to at least 100 pairs for each scene type.

5.6. Final conclusions on experiments. For the convenience of comparative descriptor analysis and making final conclusions on the work was constructed:

1) the bubble diagram (Fig.15) which clearly illustrates the number of values of the best expert rate “4” and the time costs for all descriptors, where the point size indicates the number of failed normalizations (“-1” and “0”);

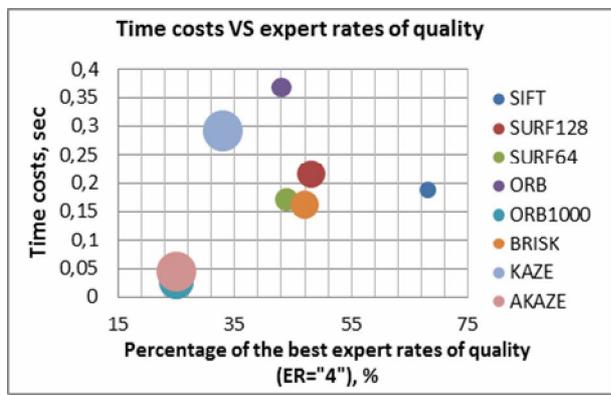


Fig. 15. Time costs VS expert rates of quality (the number of values “4” is on the abscissa axis, the time costs value is on the ordinate axis, the point size is the number of failed normalizations (“-1” and “0”), for instance, point size is 2 for SIFT and 12 for KAZE, AKAZE)

2) the rating diagram (Fig.16), which was made by converting values for each indicator into the 8-point rating scale (see paragraph 4.5). This diagram obviously presents the rating of descriptors for the considered indicators by the groups:

- quantitative estimates NP, NMO, NI;

- relative values Precision and Recall;
- time costs estimates DesT, MatchT, TotalNormT (InlierT is not given because its values are too small and their effect on time costs, in general, is not significant);

- relative quality assessments based on such expert rates: ER=“4” (the winner), ER=“-1” (absent), ER=“0” (failed) and the mean expert rate MeanER.

The diagrams 15, 16 allow to make the following conclusions:

- the descriptors AKAZE and ORB1000 have the best time costs but the least number of expert rate “4” and a high level of failed normalizations;
- the descriptors BRISK, SURF64, SIFT, SURF128 have comparable time costs and occupy an average position in time costs relative to the other descriptors, but SIFT significantly exceeds BRISK, SURF64, SURF128 in quality (a maximum number of the expert rate “4” and a minimum number of normalization failures).

- for the SIFT algorithm, the ratio of the number of found matches to the detected points is higher than in other algorithms, i.e., SIFT detects key points and gives them a description in the best way, that makes it the winner by normalization quality among the compared algorithms. As for SURF128, SURF64, they have a similar number of key points with the SIFT algorithm, but a share of found points, which are useful for normalization, is the lowest. The normalization quality with SURF128, SURF64 is similar, like with the BRISK descriptor.

- the KAZE descriptor normalizes 1/3 longer than algorithms BRISK, SURF64, SIFT, SURF128 (but ORB does it even longer), with a maximum number of normalization failures (maximum number of the rate “-1”). The total number of rates “-1” and “0” is very similar for KAZE, AKAZE;
- the ORB descriptor has the most essential time costs, and the normalization quality is comparable to SURF64, SURF128, BRISK. The great ORB time costs are caused by the significant expenses on finding matches for a large number of points (the ORB algorithm finds the maximum number of key points, and only 10% of them are used for normalization).

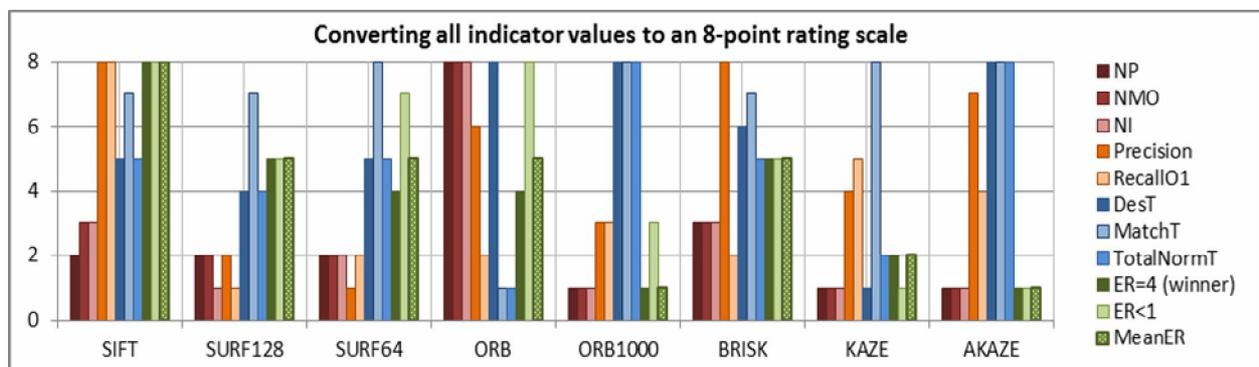


Fig. 16. Converting all indicator values to an 8-point rating scale (point 8 is the highest mark). Each value was assigned to one of the marks from 1 to 8, depending on which interval it got into (paragraph 4.5)

Conclusions

Thus, in this paper, the image normalization method implementation based on the analysis of the key points is considered in detail. Comparative estimation of the quality and time costs of the full-cycle descriptors SIFT, SURF128, SURF64, BRISK, ORB, ORB (1000), KAZE, AKAZE was conducted.

The following conclusions as to the result of the research:

- all descriptors can be used for normalization;
- the SIFT algorithm presented the best quality (even at an extreme change of lighting), however, SIFT time costs occupy an average position, along with SURF, BRISK which are significantly inferior to it by quality;
- the fastest algorithms were ORB1000 and AKAZE, the normalization quality with them is lower than with other algorithms;
- the KAZE algorithm is slower (only ORB is even more slower) and inferior to SIFT, SURF, BRISK, ORB in quality;
- the ORB descriptor has the most substantial time costs and averaged quality comparable to SURF and BRISK;
- SURF64 and SURF128 descriptors showed comparable quality and time costs, SURF64 is faster, SURF128 has better quality, but these differences are insignificant;
- the worst results were obtained for texture images;

– the symmetric method usage for searching matches at the 2nd step doubles the time costs of the normalization process in general.

Therefore, it is recommended to use the SIFT descriptor for general tasks where it is necessary to process images with the scenes, which are similar to ones of the dataset SYTOSS_NURE_pngPairs100, and the time costs requirement is not critical. If time costs are paramount and quality can be neglected within reasonable limits, it is better to use binary descriptors ORB1000 or AKAZE.

Findings on the paper are based on the image normalization of the SYTOSS_NURE_pngPairs100 dataset, which contains image scenes of different types, 40% of which are images of natural and artificial textures.

As the normalization of images of this type showed the worst results, the conclusions for the other sets may differ slightly.

In this work, the expert rates were applied to assess the normalization quality because a person can match images after normalization in the best way, but this approach to estimation is very time-costuming and is not devoid of subjectivity.

So, further research should focus on the development of an automatic criterion for assessing normalization quality and research of normalization result stability under various distortions: geometric transformation, illumination and degree of compression.

REFERENCES

1. Gorokhovatskiy, V.A. (2014), *Strukturny`j analiz i intellektual`naya obrabotka danny`kh v komp`yuternom zrenii* [Structural analysis and data mining in computer vision], Kompaniya SMIT, Khar`kov, 316 p. (in Russian)
2. Putyatin, E.P. and Averin, S.I. (1990), *Obrabotka izobrazhenij v robototekhnike* [Image processing in robotics], Mashinostroenie, Moscow, 320 p. (in Russian)
3. Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2017), "ImageNet classification with deep convolutional neural networks", *Communications of the ACM*, No60(6), pp. 84-90.
4. Putjatin E.P., Jakovleva E.V., Ljubchenko V.A. (1996), "Razlozhenie matricy centroaffinnogo preobrazovaniya dlja normalizacii izobrazhenij" [Centroaffine transformation matrix decomposition for image normalization], *Radioelektronika i informatika* No. 4 (05), pp. 91–94. (in Russian)
5. Yakovleva, E.V. (2004), "Metody` odnomerny`kh normalizacij affiny`kh preobrazovanij v zadachakh raspoznavaniya izobrazhenij: dissertation" [Methods of dimensional normalization affine transformations in image recognition tasks: dissertation], Kharkiv, 199 p. (in Russian)
6. Genkin, V.L., Erosh, I.L., and Moskalev, E.S. (1988), *Sistemy` raspoznavaniya avtomatizirovanny`kh proizvodstv* [Recognition systems for automated production], Mashinostroenie, Leningrad, 246 p. (in Russian)
7. Lyubchenko, V.A. (2004), "Matematicheskie modeli i metody` normalizacii proektivny`kh preobrazovanij v sistemakh obrabotki izobrazhenij: dissertation" [Mathematical models and methods of normalization of projective transformations in image processing systems: dissertation], Kharkiv 156 p.
8. Lowe, D.G. (2004), "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, Vol. 60. – No 2. – pp. 91-110.
9. Bay, H., Tuytelaars, T., and Van Gool, L. (2006), "Surf: Speeded up robust features", *European conference on computer vision*. Springer, Berlin, Heidelberg, pp. 404-417.
10. Rublee, E. et al. (2011), ORB: "An efficient alternative to SIFT or SURF", *2011 International conference on computer vision*, Ieee, pp. 2564-2571.
11. Leutenegger, S., Chli, M., and Siegwart, R.Y. (2011), "BRISK: Binary robust invariant scalable keypoints", *2011 International conference on computer vision*, Ieee, pp. 2548-2555.
12. Alcantarilla, P.F., Bartoli, A., and Davison, A.J. (2012), "KAZE features", *European Conference on Computer Vision*, Springer, Berlin, Heidelberg, pp. 214-227.
13. Alcantarilla, P. F., and Solutions, T. (2011), "Fast explicit diffusion for accelerated features in nonlinear scale spaces", *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 34, No 7, pp. 1281-1298.
14. Márquez-Neila P. et al. (2016), "Speeding-up homography estimation in mobile devices", *Journal of Real-Time Image Processing*, Vol. 11, No 1, pp. 141-154.

15. (2020); Detail material on “Research of descriptor based image normalization and comparative analysis of SURF, SIFT, BRISK, ORB, KAZE, AKAZE descriptors” (original image pairs, normalized images, their overlaps, the tables with different estimations, summary diagrams etc.)” available at: <https://github.com/SytossResearch/DescriptorBasedNormalization>

Надійшла (received) 15.09.2020

Прийнята до друку (accepted for publication) 11.11.2020

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Яковлева Олена Володимирівна – кандидат технічних наук, доцент кафедри інформатики, Національний технічний університет радіоелектроніки, Харків, Україна;

Olena Yakovleva – PhD, Associate Professor Department of Informatics, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine;

e-mail: olena.yakovleva@nure.ua; ORCID ID: <https://orcid.org/0000-0002-6129-6146>.

Ніколаєва Катерина Геннадіївна – інженер, SYTOSS R&D Engineer, Kharkiv, Ukraine;

Kateryna Nikolaieva – SYTOSS R&D Engineer, Kharkiv, Ukraine;

e-mail: Ekaterina.levchenko@syto.com; ORCID ID: <https://orcid.org/0000-0002-1190-6858>.

Дослідження методу нормалізації зображень на базі дескрипторів та порівняльний аналіз дескрипторів SURF, SIFT, BRISK, ORB, KAZE, AKAZE

О. В. Яковлева, К. Г. Ніколаєва

Анотація. Предметом досліджень є нормалізація зображень на основі аналізу характерних точок. Метою є розроблення математичних моделей та їх програмна реалізація для нормалізації геометричних перетворень зображень на основі аналізу дескрипторів SIFT, SURF, ORB, BRISK, KAZE, AKAZE; застосування цієї моделі для проведення порівняльного аналізу дескрипторів на основі експертних оцінок якості нормалізації, часових витрат та інших показників; формування та використання в експериментах власного датасету зі 100 реальних пар зображень, які містять сцени п'яти типів: будівлі та міські пейзажі, плоскі зображення, що зустрічаються на вулиці, плоскі зображення внутрішнього інтер'єру, природні та штучні текстури; висновки відносно використання розглянутих дескрипторів для вирішення задач нормалізації. Застосовуваними методами є: дескриптори SIFT, SURF, ORB, BRISK, KAZE, AKAZE для опису характерних точок, Nearest Neighbor Distance Ratio метод або симетричний метод для пошуку відповідних пар характерних точок з різних зображень, RANSAC метод для видалення хибних відповідностей та отримання параметрів матриці гомографії, міри подібності, програмне моделювання. Отримані результати: результати експериментів щодо нормалізації дескрипторами SIFT, SURF, ORB, BRISK, KAZE, AKAZE для 100 реальних пар зображень (нормалізовані зображення, їх перекриття, кількісні оцінки дескрипторів, precision and recall показники, часові витрати, експертні оцінки якості, конвертація всіх значень показників в 8-бальну рейтингову шкалу, підсумкові діаграми та висновки про переваги та слабкі місця дескрипторів, що порівнювалися; рекомендації відносно реалізації методу нормалізації у конкретних випадках.

Ключові слова: нормалізація геометричних перетворень; детектор; дескриптор; SURF; SIFT; BRISK; ORB; KAZE; AKAZE; метод RANSAC; метод k-найближчих сусідів; Nearest Neighbor Distance Ratio; бібліотека OpenCV.

Метод нормализации изображений на базе дескрипторов и сравнительный анализ дескрипторов SURF, SIFT, BRISK, ORB, KAZE, AKAZE

Е. В. Яковлева, Е. Г. Николаева

Анотація. Предметом исследований является нормализация изображений на основе анализа характерных точек. Целью является разработка математических моделей и их программная реализация для нормализации геометрических преобразований изображений на основе анализа дескрипторов SIFT, SURF, ORB, BRISK, KAZE, AKAZE; применение данной модели для проведения сравнительного анализа дескрипторов на основе экспертных оценок качества нормализации, временных затрат и других показателей; формирование и использование в экспериментах собственного датасета из 100 реальных пар изображений, содержащих сцены пяти типов: здания и городские пейзажи, плоские встречающиеся на улице изображения, плоские изображения внутреннего интерьера, природные и искусственные текстуры; выводы относительно использования рассмотренных дескрипторов для решения задач нормализации. Применяемыми методами являются: дескрипторы SIFT, SURF, ORB, BRISK, KAZE, AKAZE для описания характерных точек, метод Nearest Neighbor Distance Ratio или симметричный метод для поиска соответствующих пар характерных точек на разных изображениях, метод RANSAC для удаления ложных соответствий и получения параметров матрицы гомографии, меры сходства, программное моделирование. Полученные результаты: результаты экспериментов по нормализации дескрипторами SIFT, SURF, ORB, BRISK, KAZE, AKAZE для 100 реальных пар изображений (нормализованные изображения, их перекрытия, количественные оценки дескрипторов, precision and recall показатели, временные затраты, экспертные оценки качества, конвертация всех значений показателей в 8-балльную рейтинговую шкалу, итоговые диаграммы и выводы о преимуществах и слабых места сравниваемых дескрипторов; рекомендации относительно реализации метода нормализации в конкретных случаях.

Ключевые слова: нормализация геометрических преобразований; детектор; дескриптор; SURF; SIFT; BRISK; ORB; KAZE; AKAZE; метод RANSAC; метод k-ближайших соседей; Nearest Neighbor Distance Ratio; библиотека OpenCV.