STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LXII**, Number 2, 2017
DOI: 10.24193/subbi.2017.2.02

IMPROVING SIFT FOR IMAGE FEATURE EXTRACTION

RENATA DEAK, ADRIAN STERCA, AND IOAN BĂDĂRÎNZĂ

Abstract. This paper reviews a classical image feature extraction algorithm, namely SIFT (i.e. Scale Invariant Feature Transform) and modifies it in order to increase its repeatability score. We are using an approach that is inspired from another computer vision algorithm, namely FAST. The tests presented in the evaluation section show that our approach (i.e. SIFT-FAST) obtains better repeatability scores over classical SIFT.

1. Introduction

Image matching techniques can be separated into two major categories - *feature-based* or *direct*. Direct image matching techniques involve directly matching one image's pixels values to the values of the pixels of another image. Therefore, this method looks at how much the pixels of two, or more, images agree [17]. This approach can be split into two steps: finding an appropriate error metric like Mean Squared Error and deciding on an efficient search technique. Although, exhaustive search can be applied, it is not an efficient approach, in particular when dealing with high resolution images.

On the other hand, feature-based matching techniques extract interest points from the input images and aim to minimize the distance between these interest points. This method is preferred to the direct-based method as it is more robust, due to the fact that feature extraction algorithms output features that are invariant to scale, rotation and translation. One of the earliest and most notable feature extraction algorithms is SIFT - Scale Invariant Feature Transform - developed by David Lowe, first published in 1999 [2]. This enabled the development of more feature extraction methods such as SURF [8], ASIFT [11], ORB [12], BRISK [15] and FAST [18], to name a few.

Feature extraction methods have two phases. The first one is detecting the interest points in an image. There is no formal definition of what constitutes

Received by the editors: August 22, 2017.

2010 *Mathematics Subject Classification.* 68U10, 94A08.

1998 *CR Categories and Descriptors.* I.4.7 [**Image Processing and Computer Vision**]: Feature Measurement – *Feature Representation*; I.5.2 [**Pattern Recognition**]: Design Methodology – *Feature evaluation and selection.*

Key words and phrases. image feature extraction, SIFT, FAST.

an interest point, most papers defining it as an - interesting - part of an image, parts that are easily recognizable in two or more different images. This property is called repeatability, and it is used to measure the effectiveness of feature detection algorithms. It is desirable that extracted features are invariant to scale, rotation or translation. This is the reason why feature detection algorithms aim to detect corners, rather than detecting edges, as edges are invariant to translation only along their principal axis [5]. Invariance to scale is obtained by building a scale space of the input image that simulates different levels of zoom and blur applied on the initial image. The second one is building descriptors for these points, which will be used to identify a point within an image. These descriptors are later on used for matching features between images or for object detection purposes. Matching between features is done by minimizing the distance between their descriptors.

In this paper we start wih the classical SIFT algorithm and update it by adding a mechanism inspired from FAST that is meant to increase the repeatability of SIFT.

2. Related work

One of the first detectors introduced in the literature is the Harris corner detector, which detects points based on eigenvalues of the second-moment matrix [5]. However, this detector was not scale-invariant. It wasn't until Lindeberg introduced the concept of automatic scale selection [6] that the scale invariance of features was a property that feature detector algorithms pursued. In order to attach to each point its characteristic scale, Lindeberg used both the Hessian matrix and the Laplacian. What followed was a refinement of this method by Mikloajczyk and Schmidt called Harris-Laplace and Hessian-Laplace which issued robust, scale-invariant features. In order to avoid computation of the Laplacian of Gaussians, the idea to approximate this by the Difference of Gaussians was introduced by David Lowe, idea that was later used in his implementation of SIFT [2].

In terms of feature descriptors, there has been a wide variety of methods introduced, such as Gaussian derivatives [7], complex features [9] [10] and descriptors such as SIFT [2] that capture information about the spatial intensity patterns in the neighbourhood of the interest point. To date, this descriptor has proved to be the most robust one. Although there have been alternatives, to improve performance - such as PCA-SIFT, which encodes information in a 36-dimensional vector [19] - they have proven to be less distinctive [14]. GLOH was proposed as an alternative [14], which has proved to be more distinctive, but it is computationally expensive.

Another alternative to the SIFT descriptor was proposed by Se et al. [13], which is both fast and distinctive enough, however it has a drawback in the fact that the vectors extracted are of high dimensions, making the matching phase more difficult.

In [8] we can find another feature extraction method, named SURF (Speeded Up Robust Features), which defines three main steps for searching for discrete image correspondences: detection step, description step and matching step. In the first step, they are selecting distinctive locations in the image, like blobs, corners and T-junctions. These distinctive locations are called 'interest points'. In the second step, a feature vector is defined for each neighbourhood of every interest point. These vectors are used further in step three where they are matched between different images, by computing the Mahalanobis or Euclidean distance. Even though SURF is know for its robustness and speed. there are other algorithms, like BRISK (Binary Robust Invariant Scalable Keypoints) [15] that can achieve comparable quality for matching but with much less computation time. BRISK is a method for generating keypoints from an image in two phases: detecting scale-space keypoints using a saliency criterion and keypoint description.

ORB (Oriented FAST and Rotated BRIEF) is another very fast binary descriptor that is build on top of FAST keypoint detector and BRIEF descriptor. Combining these two methods, you can achive very good performance and very low cost. Our own approach combines the robustness of the SIFT descriptors with the detection of the FAST algorithm in order to achieve finding better candidate points from the first extraction of points, but also increasing the repeatability of the features extracted.

3. SIFT - Scale Invariant Feature Transform

The SIFT algorithm outputs features that are invariant to scale, rotation and translation. The first step in SIFT feature extraction is to compute a Gaussian scale-space from the initial image. 5 octaves of images are constructed from the initial image where each octave contains variants of the initial picture with a decreased samplerate. In each octave there are 5 images with the same samplerate, but with increasing blur levels (Gaussian blur is used). A part of this Gaussian scale-space is depicted in Fig. 1 for an image that is later used in the evaluation section. After this, the DOG (i.e. Difference of Gaussian) space is computed by applying the differential operator to the Gaussian scale-space and the 3D extremum points are extracted from the DOG space. Similarly, a part of the DOG representation is displayed in Fig. 2 for the same image. The 3D extremum points are coarsely detected from the

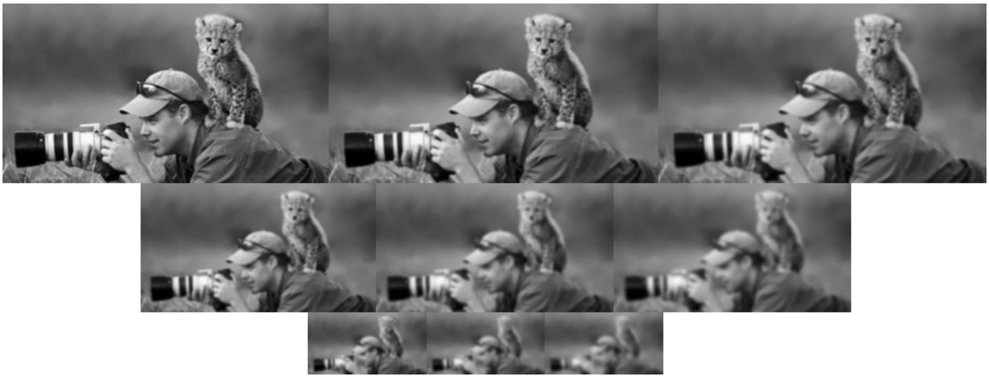

Figure 1. Gaussian scale space. The first row represents the first 3 images in the first octave, the second row contains the first 3 images from the second octave, and the third from the third octave.

DoG scale-space by taking the minimum or the maximum from a neighbourhood of 26 pixels (see Fig. 3). This is done in order to have extremum points that are scale invariant. Unfortunately, these extremum candidate points are very are sensitive to noise. Several filters are applied then in order to discard low importance extreme points like low contrast keypoints or candidate keypoints that are on the edges. Furthermore, maximum or minimum points are not situated directly on the pixel, they usually lie in between pixels. So in order to be able to address such a pixel, a position refinement is applied, as well as a scale refinement, since points detected previously are constrained to the sampling grid. After low-contrasted pixels are discarded, local interpolation (using an approximation of the second order Taylor polynomial) is applied on the remaining candidate points to refine their location and scale.

The final step in SIFT is constructing a 128 byte descriptor for each keypoint found. SIFT feature descriptors are computed by extracting one or more dominant orientations for each keypoints over a normalized patch. This ensures the rotation-invariant property of the extracted keypoints. Due to the fact that there may be more than one dominant orientation for a single keypoint, the number of feature descriptors extracted may be higher than the number of features detected in the previous steps. The first step in extracting the dominant orientation is to build a patch around the keypoint which contains pixels that lie at a distance smaller than a threshold value from the keypoint's position. Then, within the normalized patch, a magnitude and orientation are computed from the gradient of the image with respect to the x-coordinate and

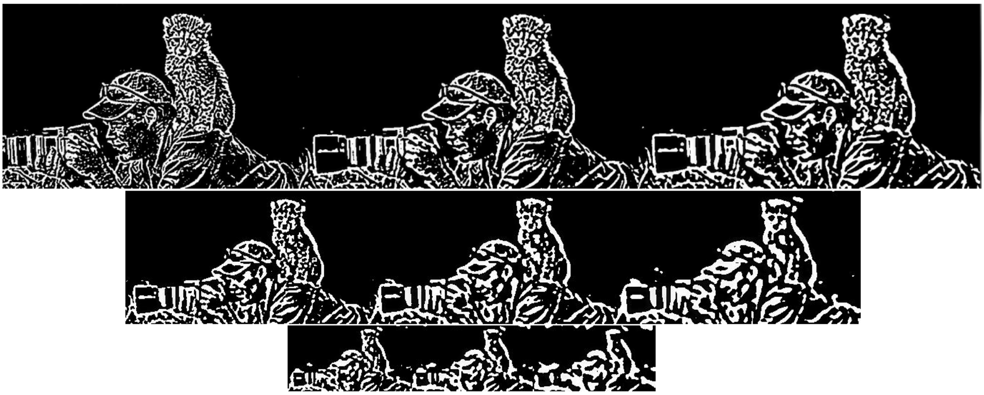

FIGURE 2. The DoG representation. The first row represents the first 3 images in the first octave, the second row contains the first 3 images from the second octave, and the third from the third octave.

the y-coordinate. A histogram of orientations will be built from the orientations extracted. The histogram is built by dividing the interval $[0, 2\pi]$ into 36 bins of 10 degrees each. Then the orientation is assigned to the closest bin. For example, if the orientation is $\pi/4$, the corresponding bin will be the one for degrees 40-49. In order to smooth out noise and make distant pixels have a smaller influence on the gradient assignation, the entries are weighted by a Gaussian window function which is centered at the interest point. The next step in extracting dominant orientations is to smooth the histogram by applying a circular convolution six times with a three-tap box filter. From the smoothed histogram, the orientations will be extracted from local maxima positions that are larger 0.8 times than the global maximum. Consequently, we may extract more than one orientation for a single interest point. Once the orientations have been computed for each keypoint, this information is quantized into 128 dimension vectors. In order to compute the descriptor, the information of the local spatial distribution of the gradient orientation on a particular neighborhood must be encoded. As a neighborhood, there have been papers where the entire image was used [1]. However, the original SIFT method takes a square normalized patch aligned with the orientation of the point, to induce invariance to scale, rotation and translation [2].

[1] Image taken from http://www.aishack.in/tutorials/sift-scale-invariant-feature-transform-keypoints/

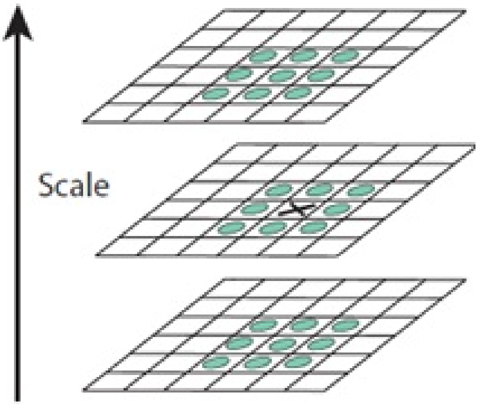

Figure 3. DoG 3D extremum candidate, if pixel 'x' is either smaller or larger than all its 26 neighbours. The point marked with X in the middle image represents the candidate point, the image below is the image with a lower blur level and the image above is the image with higher blur level [1]

4. Improving the repeatability of SIFT

In the classical SIFT implementation candidate points are extracted by finding 3D extrema points, in neighborhoods of 27 pixels. This means that each pixel of the image will be compared to its 26 neighbors - 9 from the image with a smaller level of blur, 8 from the current image and 9 from the image with a greater level of blur. A pixel is considered a candidate point if its value is either smaller or larger than all of its 26 neighbors. Our new approach is to use FAST [3] for detecting candidate keypoints of the DoG scale-space. FAST is a corner detection algorithm and in FAST, a circle of sixteen pixels is also known as the Bresenham circle of radius 3 [20] - around a candidate pixel p is considered. The pixel is a corner if there is a consecutive sequence of n pixels in the circle which are all either brighter than the candidate pixel by a certain threshold or darker than the candidate point by the same threshold. In our approach for increasing the repeatability of SIFT, instead of searching through a square of $3x3$ neighbors in 3 dimensions, the search for extrema is done by searching through a circle of radius 3 of 16 pixels a sequence of n pixels that are all either brighter than the candidate point by a certain threshold, or are darker than the candidate point by the same threshold (see Fig. 4).

The motivation behind choosing this combination of methods is the fact that this detection has the potential of extracting keypoints with high repeatability score. The repeatability is a desired property of the extracted features, as it evaluates whether or not the feature will be detected in other images containing the same scene. The FAST detection shows great potential towards this goal, since it does not restrict the pixel to be brighter or darker than all its nearest 26 neighbors. Instead, a circle is considered, from which a sequence of n pixels must be either brighter or darker than the candidate point. So the chances that the feature might be detected in a blurrier image, for example, are higher than with the classical SIFT detection. For the threshold t, tests have been run in order to evaluate how it affects the repeatability score of the extracted features, and the best value obtained was 0.018. Based on the tests from [16], the value for n was left to 12, as it provides the best results in terms of number of features extracted and the redundancy of the extracted features. This means that from each of the three images a sequence of 12 consecutive circle pixels is found, leading to 36 pixels that should be either darker or brighter than the candidate pixel by a threshold t.

5. Evaluation

In order to evaluate our SIFT-FAST approach, we have implemented classical SIFT and SIFT-FAST and compared the two algorithms on photos from different domains (thus having different entropy levels): a photo with an animal in nature (containing a reasonable amount of color changes and large blurred areas in the background), a human face (containing distinct areas of color changes) and a landscape photo (containing many small areas with small color fluctuations - i.e. the grass). The dimensions of these 3 photos are also different: 915x497, 500x366 and respectively, 425x378 pixels. We have compared the two implementations using two metrics: *the computation time* and *the repeatability score*. The computation time metric is evaluated for both approaches in order to see how much computational overhead does SIFT-FAST introduce and the second metric, repeatability, is used to evaluate if SIFT-FAST is more suited than classic SIFT for object recognition in images. For the first metric, considering that the only difference between these implementations is the extraction of the initial set of keypoints, the time required to extract these points was measured for both approaches and the results obtained are in figures 5, 6 and 7.

As it can be observed in these figures, classical SIFT detection is more efficient from the point of view of the execution time, compared to SIFT-FAST. This is due to the fact that for SIFT-FAST, a consecutive sequence of 12 pixels needs to be found in three different images, whereas with SIFT the coordinates of the pixels that are used in the comparison are known beforehand. However, this is not a big drawback for the SIFT-FAST approach, as it is still executed

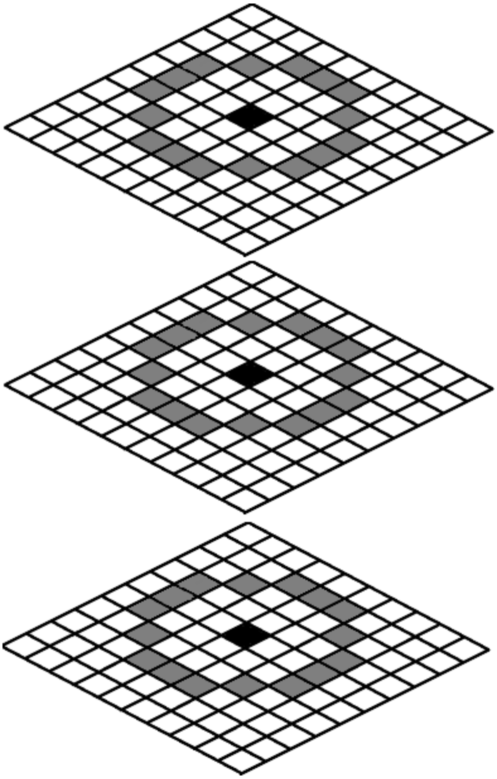

Figure 4. Candidate point detection for FAST-SIFT keypoint. The three grids represent 3 consecutive scales of an image within an octave. The black pixel is the candidate point and the grey pixels represent the circle from which a sequence of n consecutive pixels needs to be darker or brighter than the black pixel by a certain threshold.

under 0.6 seconds. It just means with SIFT the check is completed faster. Although SIFT-FAST takes more time because it extracts more reliable points in this first step than classical SIFT.

Another important property of feature detecting algorithms is the *repeatability of the features*. This means that having two different images of the same scene, the features detected in the first image are detected in the second image as well. The repeatability score for an image was computed by taking the image and a blurrier version of the same image (obtained by applying a Gaussian blur filter) and computing the percentage of keypoints that are found in both images (i.e. percentage of common keypoints).

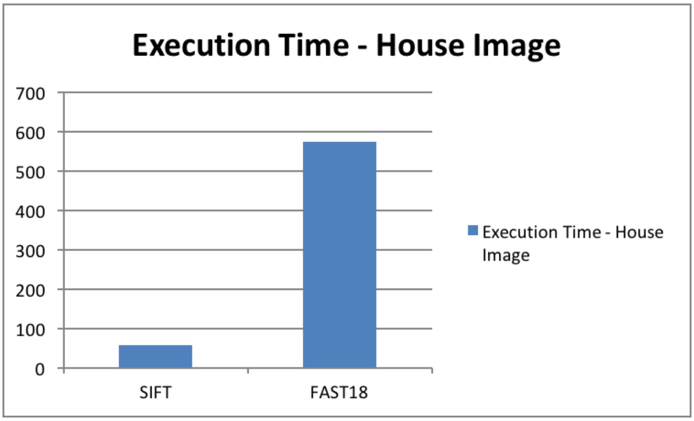

FIGURE 5. Execution time for extracting first set of keypoints for an image with a house. FAST18 represents running FAST detection with threshold=0.018. (time is measured in milliseconds)

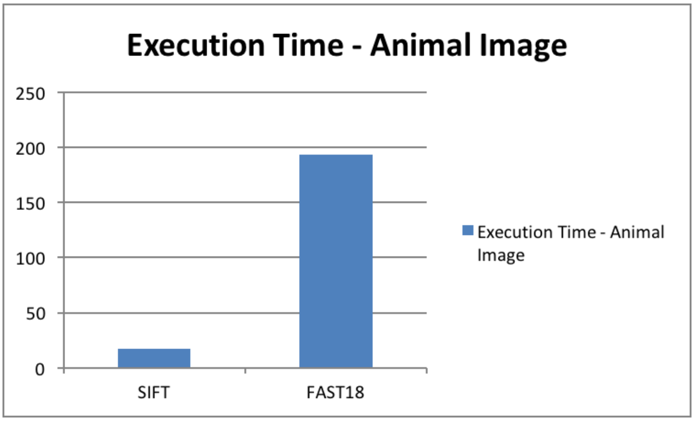

FIGURE 6. Execution time for extracting first set of keypoints for the image with an animal. (time is measured in milliseconds)

The results obtained for the same set of input images are illustrated in figures 8, 9, and 10, respectively. The notations FAST16, FAST17, FAST18 and FAST19 represent the SIFT-FAST algorithm with the value 0.016, 0.017, 0.018 and 0.019 for the threshold used for comparisons. From these tests, the conclusion is that overall the most suitable value for the threshold is 0.018.

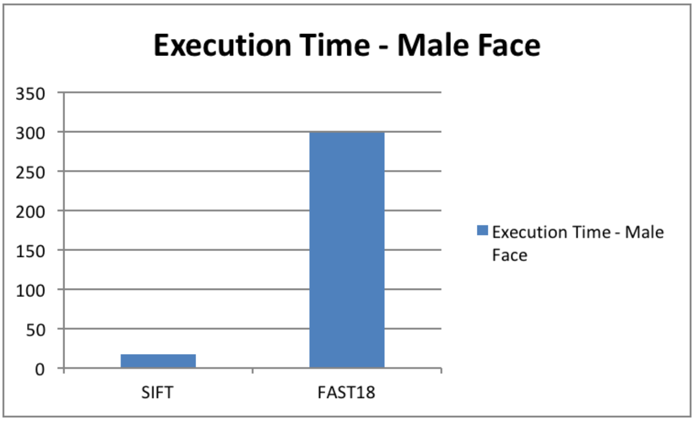

Figure 7. Execution time for extracting first set of keypoints for an image with a male face. (time is measured in milliseconds)

It is clearly visible in figures 8 - 10 that the SIFT-FAST approach yields better repeatability results than the classical SIFT algorithm irrespective of the threshold used, although some threshold values give better results than others. This result is ensured by the initial extraction of the keypoints. If a point is chosen as candidate keypoint in one image, in means that it has found the sequence of 12 pixels that are either brighter or darker in the current image, in an image with a smaller level of blur and an image with a higher level of blur. Consequently, there is a high probability that if a pixel was chosen as candidate in one image, it will be found as candidate in another image of the same scene (with different blur level, luminosity or small translate transformations applied).

The advantages of the SIFT-FAST approach over classical SIFT can also be depicted visually. In Fig. 11 we show the initial set of candidate points extracted from the three images using classical SIFT. As it can be seen, in this first phase of feature detection, the features are scattered all over the image, and they are not reliable in this phase. Then, following the SIFT workflow, a number of filters are applied to this initial set of candidate points: threshold filter, low-contrast filter, quadratic interpolation and removal of keypoints located on edges. The final set of keypoints is depicted in Fig. 12. Fig. 13 illustrates this first initial set of keypoints obtained by our SIFT-FAST approach in the same three images. It can be observed that unlike the initial detection with classical SIFT, the keypoints are much better positioned. For example in the first image, there is no point selected in the background, where

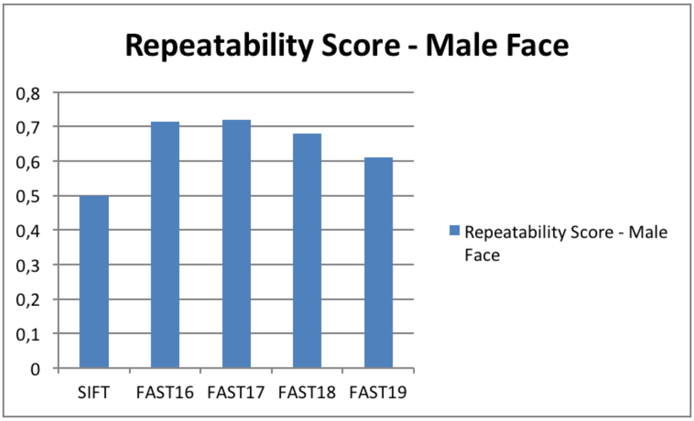

FIGURE 8. Repeatability score for image with male face.

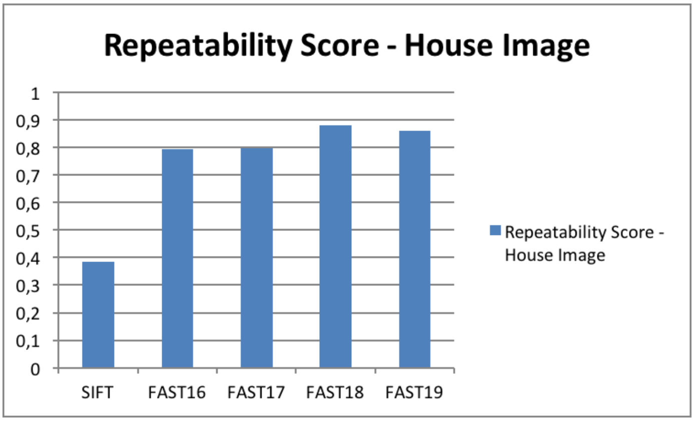

FIGURE 9. Repeatability score for image with house.

the image is out of focus, and even with the human eye, no object can be uniquely distinguished, there are no keypoints detected. With classical SIFT initial detection, the points where scattered all over the image, as it can be observed in figure 11. Then, the workflow continues the same as for classical SIFT algorithm and the final set of keypoints extracted with this approach is represented in Fig. 14. Comparing the final keypoints extracted by this approach and the classical SIFT approach it can be observed that this approach extracts fewer candidate keypoints than SIFT. However, defining what makes an extracted keypoint important is highly dependent on the application domain. For example, by comparing the points extracted for the first

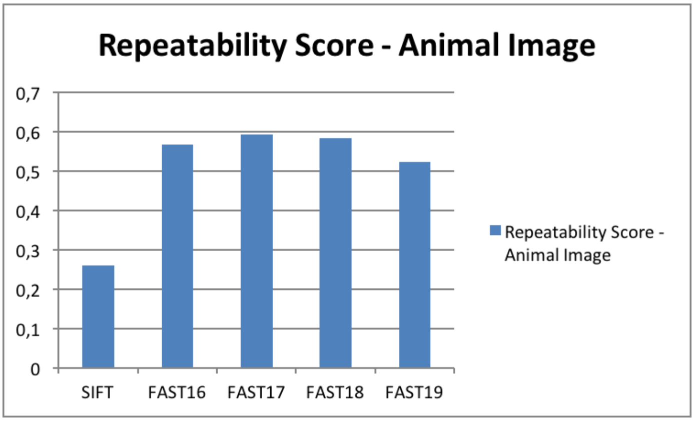

Figure 10. Repeatability score for image with animal.

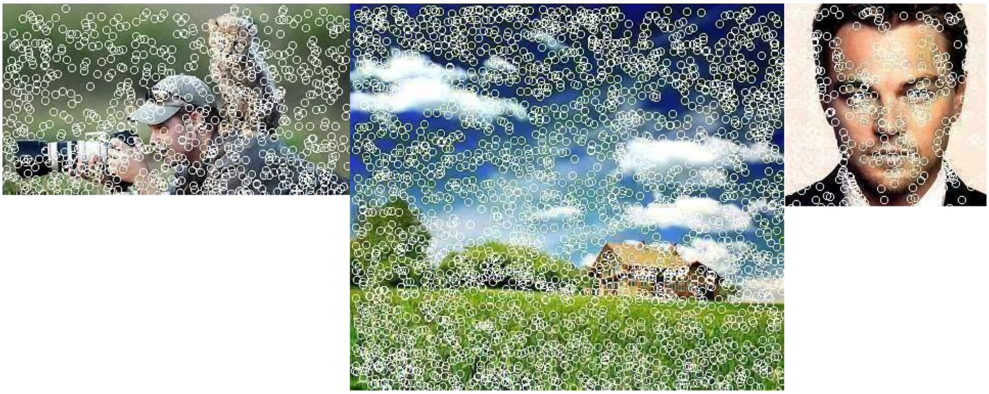

Figure 11. The first set of keypoints detected in our test images using classical SIFT.

images in the three test images by SIFT and the SIFT-FAST approach, it can be observed that the latter extracts little to no points on the man's shirt, which seems correct as the variations of contrast in that area of the image are generated by shadow only.

6. Conclusions and future work

We considerred in this paper the SIFT feature extraction algorithm introduced by David Lowe in 1999 [2]. A new approach was proposed by combining the classical SIFT algorithm with a FAST-like detection of initial keypoints.

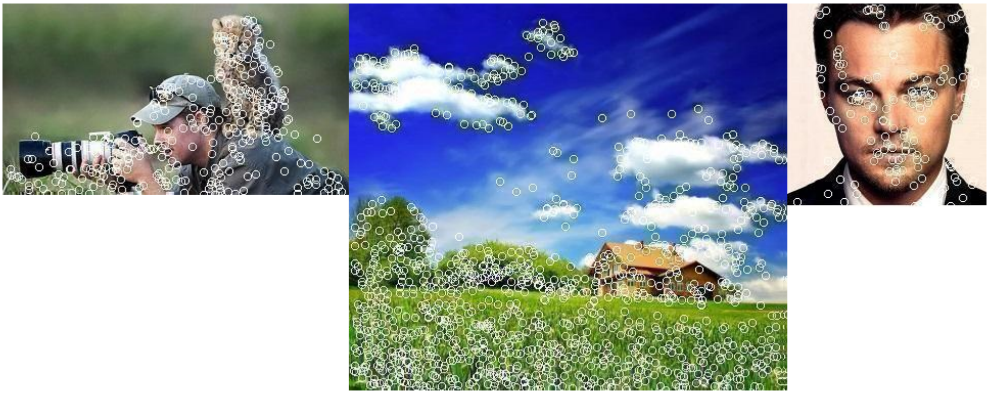

Figure 12. The final set of keypoints detected in our test images using classical SIFT (after low-contrast filtering, interpolation)

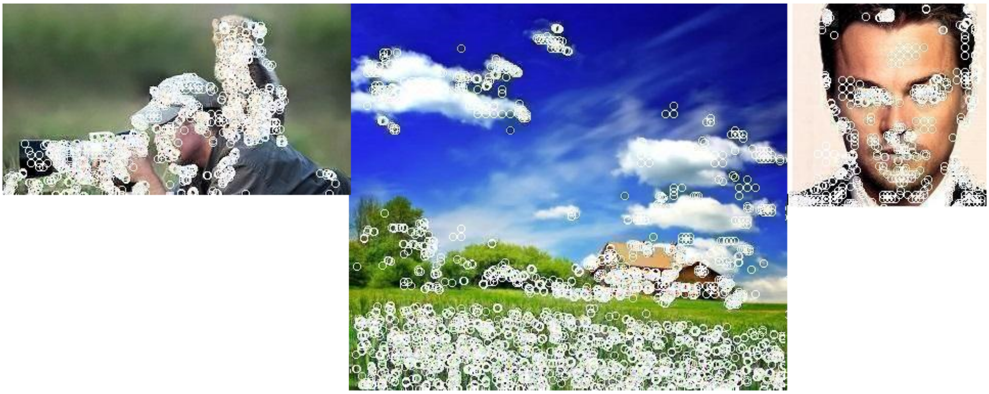

Figure 13. The first set of keypoints detected in our test images using SIFT-FAST approach.

Instead of scanning for 3D local extrema points using the 26 neighbors as the SIFT algorithm does, the points are scanned in a FAST-like way. That is, the candidate point is selected if on the circle of radius 3 having the point in its center, there are 12 pixels either brighter or darker than the point by a threshold. This check is done on the current level of blur, on the previous sample image with a smaller level of blur and on the next sample image with a higher level of blur.

In our tests, we compared these two methods in terms of execution time and repeatability of features. We showed that although using the FAST detector

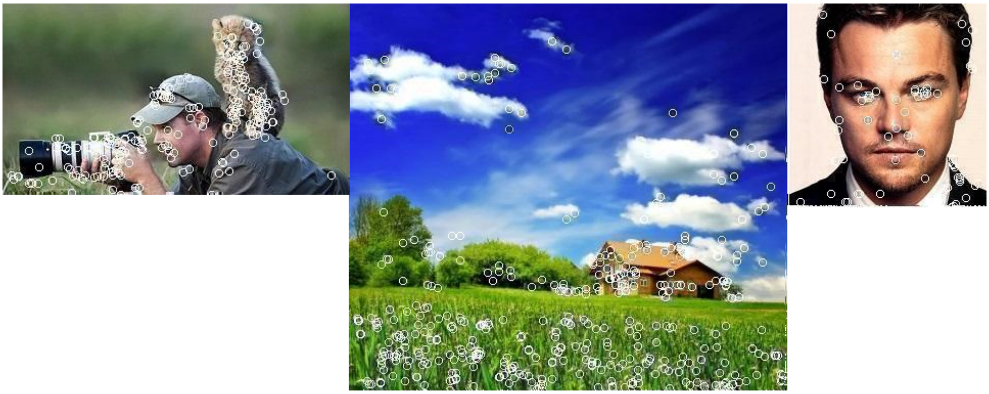

FIGURE 14. The final set of keypoints detected in our test images using SIFT-FAST approach (after low-contrast filtering, interpolation)

for extracting the initial set of keypoints was more time-consuming, it yielded better results in terms of repeatability of features which is important for image object recognition tasks. The reason why execution time is higher using SIFT-FAST algorithm is that a consecutive sequence of 12 pixels needs to be found, on three levels of blur, whereas for the classical SIFT, there are 26 neighbors that are checked only. However, the fact that the repeatability test had better results of this approach, than the classical SIFT, makes this drawback have lesser importance.

REFERENCES

[1] Hassner, T., Mayzels, V., Zelnik-Manor, L., On SIFTs and their scales, IEEE Conference on Computer Vision and Pattern Recognition, Washington DC, USA, June 16-21, 2012, pp. 1522-1528.

[2] Lowe, D., Object recognition from local scale-invariant features, In Proceedings of the 7th International Conference on Computer Vision, Washington DC, USA, September 20-25, 1999, pp. 1150-1157.

[3] Rosten, E., Drummond, T.: Fusing points and lines for high performance tracking, 10th IEEE International Conference on Computer Vision, Washington DC, USA, October 17-20, 2005, pp. 1508-1515.

[4] Otero, I.R., Delbracio, M., The anatomy of the SIFT method, Image Processing On Line, vol. 4, 2014, pp. 370-396.

[5] Harris, C., Stephens, M.: A combined corner and edge detector, Proceedings of the 4th Alvey Vision Conference, Manchester, 31 August - 2 September, 1988, pp. 147-151.

[6] Lindeberg, T., Scale-space theory in computer vision, Kluwer Academic Publishers Norwell, MA, USA,1994.

[7] Florack, L.M.J., Haar Romeny, B.M.T., Koenderink, J.J., Viergever, M.A.: General intensity transformations and differential invariants, Journal of Mathematical Imaging and Vision, May 1994, Volume 4, Issue 2, pp 171-187.

[8] Bay, H., Tuytelaars, T., Van Gool, L., Surf: Speeded up robust features., Proceedings of the European Conference on Computer Vision, Graz, Austria, May 2006, pp 404-417.

[9] Baumberg, A., Reliable feature matching across widely separated views, Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, 15 June 2000, pp. 774-781.

[10] Schaffalitzky, F., Zisserman, A., Multi-view matching for unordered image sets, or "How do I organize my holiday snaps', European Conference on Computer Vision, Copenhagen, Denmark, May 28-31, 2002, pp. 414-431.

[11] G. Yu and J-M. Morel, ASIFT: An Algorithm for Fully Affine Invariant Comparison, Image Processing On Line, vol. 1, 2011, pp. 438-469.

[12] Rublee, E., Rabaud, V., Konolige, K., Bradski, G., ORB: An efficient alternative to SIFT or SURF, Proceedings of IEEE International Conference on Computer Vision, Washington DC, USA, November 06-13, 2011, pp. 2564-2571.

[13] Se, S., Ng, H., Jasiobedzki, P., Moyung, T., Vision based modeling and localization for planetary exploration rovers, Proceedings of the 55th International Astronautical Congress, Vancouver, Canada, 4-8 October pp. 364-375.

[14] Mikolajczyk, K., Schmid, C., A performance evaluation of local descriptors. Pattern Analysis and Machine Intelligence, vol. 27, issue 10, 2005, pp. 1615-1630.

[15] Leutenegger, S., Chli, M., Siegwart, R.Y., BRISK: Binary Robust Invariant Scalable Keypoints, Proceedings of IEEE International Conference on Computer Vision, Barcelona, Spain, 6-13 November, 2011, pp. 2548-2555.

[16] Rosten, E., Porter, R., Drummond, T., Faster and better: A machine learning approach to corner detection, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 32, issue 1, pp. 105-119, 2010.

[17] Szeliski, R., Image alignment and stitching: a tutorial, Foundations and Trends in Computer Graphics and Computer Vision, Now Publishers, pp. 1-104, 2006.

[18] Rosten E., Drummond, T., Machine learning for high-speed corner detection, European Conference on Computer Vision, Graz, Austria, May 07-13, 2006, pp. 430-443.

[19] Ke, Y., Sukthankar, R., PCA-SIFT: A more distinctive representation for local image descriptors, IEEE Conference on Computer Vision and Pattern Recognition, Washington DC, USA, 27 June-2 July 2004.

[20] Donald Hearn, M. Pauline Baker, Computer graphics, Prentice-Hall, USA, 1994.

Faculty of Mathematics and Computer Science, Babeş–Bolyai University, Cluj-Napoca, Romania

E-mail address: `drhp0888@scs.ubbcluj.ro`
E-mail address: `forest@cs.ubbcluj.ro`
E-mail address: `ionutb@cs.ubbcluj.ro`