

A radial keypoint detector

Jasna Maver

Department of Library and Information Science and Book Studies, Faculty of Arts, University of Ljubljana, Aškerčeva 2, 1000 Ljubljana, Slovenia

E-mail: jasna.maver@ff.uni-lj.si

Abstract. The work presents a radial keypoint detector, a new version of one of the self-similarity detectors described in [14]. Several improvements are proposed to make the detector more efficient. We extend the detected keypoints with a local region orientation allowing the keypoint invariance to the image plane rotation. The radial detector is compared with the DoG detector at the theoretical level. In the experimental evaluation, the radial detector is tested on full image sequences of the HPatches dataset. The radial detector gives competitive repeatability and matching scores compared to the DoG detector which is one of the most popular keypoint detectors.

Key words: local feature detector, dependent effects model, keypoint detector evaluation

Radialni detektor značilnih točk

V delu predstavimo Radialni detektor, novo verzijo enega od detektorjev samopodobnih značilnih točk opisanih v delu [14]. Predlagamo več izboljšav, ki zagotovijo večjo učinkovitost detektorja. Detektirane značilne točke razširimo z orientacijo lokalne regije, kar omogoča invariantnost na rotacijo slike. Na teoretični ravni primerjamo Radialni detektor z detektorjem DoG. Radialni detektor eksperimentalno ovrednotimo na podatkovni bazi HPatches slikovnih sekvenc. Rezultati, ki jih dobimo za merjeno ponovljivost in ujemanje na podlagi opisnika značilnih točk, so konkurenčni rezultatom, dobljenimi z detektorjem DoG, ki je eden najbolj priljubljenih detektorjev značilnih točk.

1 INTRODUCTION

A feature or keypoint detector is an algorithm which identifies salient locations or keypoints in images. An object or scene appearance in images changes due to different conditions when capturing images, e.g. a change in illumination and a viewpoint or due to different image deformations, e.g. a lossy image compression. We expect that keypoints repeat in images of the same scene. Different measures based on the image intensity in a local neighbourhood of a keypoint form different descriptors. Intensity variations near the keypoints should make descriptors easily distinguished. Many computer vision applications, such as camera calibration, scene reconstruction [21], [6], object or scene recognition [19], [15], [10], panorama creation [7], [4], robot navigation [5], and object tracking rely on keypoint matching.

Computer vision researchers have developed many different keypoint detectors. An extensive survey is

made by Tuytelaars and Mikolajczyk in [22]. The Kadir and Brady salient region detector [8] is inspired by the information theory. The saliency of the location corresponds to local complexity or unpredictability. SIFT [9] detects blob-like keypoints. They are local extrema in the scale-space obtained by image filtering with differences of Gaussians. SURF [3], based on the Hessian matrix, approximates the Gaussian filters with box filters which enables fast computation of saliency maps using integral images, thus speeding up the keypoint detection. The traditional Gaussian scale-space approach has its limitations since it blurs the noise and fine details to the same degree, reducing localization accuracy and distinctiveness. The KAZE detector [1] uses nonlinear diffusion filtering to encourage smoothing within a region instead of smoothing across boundaries. MSER or Maximally Stable Extremal Regions proposed by Matas et al. [13] detects regions where all pixels inside the region have either higher or lower intensity than those on its outer boundary. It belongs to a group of affine region detectors, where we can find also the Harris and Hessian affine [16] region detectors. SUSAN [20] is a simple and efficient corner detector. It computes the fraction of pixels within a neighbourhood, which have a similar intensity to the centre pixel. Corners can then be localized by thresholding this measure and selecting local minima. The FAST detector [18] builds on the SUSAN detector. FAST evaluates only 16 pixels on a circle of radius $r = 3$. If a set of the N contiguous pixels in the circle are all brighter or all darker than the intensity of the candidate pixel, then the candidate pixel is classified as a corner. The algorithm computes the decision tree by learning the distribution of the corner configuration

from a training set of a specific environment. Unlike FAST, the AGAST detector proposed by Mair et al. [12] uses a binary decision tree which is generic and does not have to be adapted to new environments. The self-similarity detectors, proposed in [14], exploit three complementary parts of the variance computed on a local region. The three proposed saliencies, i.e. the radial, tangential, and residual, are invariant to rotation, photometric shift, and the magnitude of the local region contrast, they are covariant with translation and scaling, and are robust to intra-class variations. The detectors did not get much popularity mainly due to a low computational speed, though a hardware solution would enable fast computation. The algorithm also does not provide the orientation of a local region. The computation of the radial saliency alone can be speeded-up and in this paper we explain the steps taken to achieve this goal. We also use additional techniques to improve the quality of the keypoint detection and extend the keypoints with a local region orientation.

2 METHOD

The radial detector uses α effects which are one kind of effects of the dependent effects model (DEM).

2.1 Local region represented by DEM

Circular local region \mathcal{P} is transformed into rectangular patch P by means of polar raster sampling. Samples are taken at N angles, $\phi_n = \frac{2\pi n}{N}; n = 0, 1, \dots, N-1$, and M circles with radii $r_m = m; m = 0, \dots, M-1$. The intensity value from location (r_m, ϕ_n) in \mathcal{P} is transformed to the m -th column and n -th row of P , where we denote it as $I_{mn} = I(m, n)$, as shown in Fig. 1. Rectangle P is decomposed by DEM in the

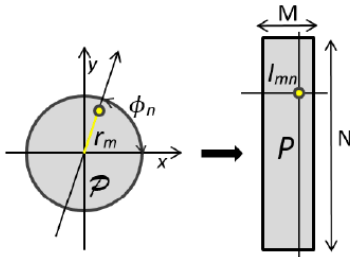


Figure 1. Polar raster sampling of circular local region \mathcal{P} .

following way. Let C_m and R_n denote the sums of intensity values from the m -th column and n -th row of P , respectively; i.e., $C_m = \sum_{n=0}^{N-1} I_{mn}$, and $R_n = \sum_{m=0}^{M-1} I_{mn}$. The dependent effects model represents P with three types of effects: radial or column effects α_m , tangential or row effects β_n , and interaction effects γ_{mn} , where $m = 0, \dots, M-1; n = 0, \dots, N-1$. Each element I_{mn} of P can be written as the following linear

combination

$$I_{mn} = \bar{I} + \alpha_m + \beta_n + \gamma_{mn},$$

where \bar{I} denotes the mean value of P , i.e.,

$$\bar{I} = \frac{1}{MN} \cdot \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I_{mn},$$

while α_m and β_n denote deviations of the row and column means from \bar{I} :

$$\begin{aligned} \alpha_m &= \frac{1}{N} \cdot C_m - \bar{I}, \\ \beta_n &= \frac{1}{M} \cdot R_n - \bar{I}. \end{aligned} \quad (1)$$

Interaction effects γ_{mn} are therefore computed as:

$$\gamma_{mn} = I_{mn} - \bar{I} - \alpha_m - \beta_n.$$

2.2 Local region saliency

The radial detector uses α -effects of DEM. The total variation of P represented by the total sum-of-squares

$$V_{P_{M \times N}} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (I_{mn} - \bar{I}_{M \times N})^2$$

can be described as a sum of two component variations $V_{P_{M \times N}} = W_{P_{M \times N}} + B_{P_{M \times N}}$. Here $W_{P_{M \times N}}$ tells how much variation is there within each of the columns of P , with columns being represented by the columns means, while $B_{P_{M \times N}}$ tells how much variation is there between the means of the columns of P . For a reader who is unfamiliar with this technique, Fig. 3 illustrates an example. When the variation within the columns of P is small, P can be well approximated by means of its columns. Remember that columns of P correspond to circles of \mathcal{P} (see Fig. 2). Saliency is defined by the ratio:

$$S_{M \times N} = \frac{B_{P_{M \times N}}}{V_{P_{M \times N}}} \quad (2)$$

with

$$B_{P_{M \times N}}(M, N) = N \sum_{m=0}^{M-1} \alpha_m^2.$$

The values of saliency (2) lie on an interval from 0 to 1. Value 0 represents the situations where all variation of P is due to the variation within the columns, while value 1 the situations where all variation is due to the variation between columns.

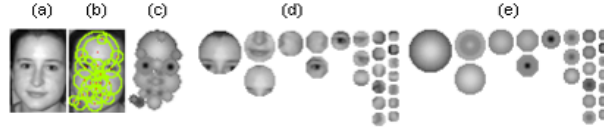


Figure 2. Blob features obtained by the radial detector. The radial detector approximates local regions with the averages computed for circles. (a) Original image with resolution (66×72) pixels, (b) Green circles denote detected features obtained for the region sizes between 4 to 12 circles, (c) Reconstruction obtained with the feature approximations. The larger features are drawn before the smaller and the less salient before the more salient. (d),(e) Features and their approximations.

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---------------|---------------|---------------|---|--|---|---|---|--|---|---|---|--|---|---|---|--|---------------|---------------|---------------|--|--|--|
| <table style="display: inline-table; border-collapse: collapse;"> <tr> <td style="padding: 0 10px;">m</td> <td style="padding: 0 10px;">0</td> <td style="padding: 0 10px;">1</td> <td style="padding: 0 10px;">2</td> </tr> <tr> <td style="padding: 0 10px;"></td> <td style="padding: 0 10px;">6</td> <td style="padding: 0 10px;">1</td> <td style="padding: 0 10px;">3</td> </tr> <tr> <td style="padding: 0 10px;"></td> <td style="padding: 0 10px;">6</td> <td style="padding: 0 10px;">3</td> <td style="padding: 0 10px;">5</td> </tr> <tr> <td style="padding: 0 10px;"></td> <td style="padding: 0 10px;">6</td> <td style="padding: 0 10px;">2</td> <td style="padding: 0 10px;">4</td> </tr> <tr> <td style="padding: 0 10px;"></td> <td style="padding: 0 10px;">$\bar{I}_0=6$</td> <td style="padding: 0 10px;">$\bar{I}_1=2$</td> <td style="padding: 0 10px;">$\bar{I}_2=4$</td> </tr> </table> | m | 0 | 1 | 2 | | 6 | 1 | 3 | | 6 | 3 | 5 | | 6 | 2 | 4 | | $\bar{I}_0=6$ | $\bar{I}_1=2$ | $\bar{I}_2=4$ | | | |
| m | 0 | 1 | 2 | | | | | | | | | | | | | | | | | | | | |
| | 6 | 1 | 3 | | | | | | | | | | | | | | | | | | | | |
| | 6 | 3 | 5 | | | | | | | | | | | | | | | | | | | | |
| | 6 | 2 | 4 | | | | | | | | | | | | | | | | | | | | |
| | $\bar{I}_0=6$ | $\bar{I}_1=2$ | $\bar{I}_2=4$ | | | | | | | | | | | | | | | | | | | | |
| $\bar{I} = \frac{6+6+6+1+3+2+3+5+4}{9}$ | | | | | | | | | | | | | | | | | | | | | | | |
| $= \frac{36}{9} = 4$ | | | | | | | | | | | | | | | | | | | | | | | |
| $V_P = W_P + B_P$ | | | | | | | | | | | | | | | | | | | | | | | |
| $28 = 4 + 24$ | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|-------------------------------------|-------------------------------------|
| $V_P = (6-4)^2 + (6-4)^2 + (6-4)^2$ | $V_P = (6-4)^2 + (6-4)^2 + (6-4)^2$ |
| $+ (1-4)^2 + (3-4)^2 + (2-4)^2$ | $+ (1-2)^2 + (3-2)^2 + (2-2)^2$ |
| $+ (3-4)^2 + (5-4)^2 + (4-4)^2$ | $+ (3-4)^2 + (5-4)^2 + (4-4)^2$ |
| $= 4+4+4+3+4+4+1+4+0 = 28$ | $= 0+0+0+1+1+0+1+1+0 = 4$ |
| $W_P = (6-6)^2 + (6-6)^2 + (6-6)^2$ | $B_P = (6-4)^2 + (6-4)^2 + (6-4)^2$ |
| $+ (1-2)^2 + (3-2)^2 + (2-2)^2$ | $+ (2-4)^2 + (2-4)^2 + (2-4)^2$ |
| $+ (3-4)^2 + (5-4)^2 + (4-4)^2$ | $+ (4-4)^2 + (4-4)^2 + (4-4)^2$ |
| $= 0+0+0+1+1+0+1+1+0 = 4$ | $= 4+4+4+4+4+4+0+0+0 = 24$ |

Figure 3. Computation of V_P and its two parts W_P and B_P .

2.3 Keypoint detection

For the keypoint detection, a stack of saliency maps is computed:

$$\begin{aligned}
 S(x, y, m) &= \frac{N \sum_{i=0}^{m-1} \alpha_i^2(x, y)}{V_{P_{m \times N}}(x, y)} \\
 &= \frac{\sum_{i=0}^{m-1} (C_i(x, y) - \bar{C}(x, y))^2}{NV_{P_{m \times N}}(x, y)} \quad (3) \\
 & \quad m = 0, \dots, M.
 \end{aligned}$$

Since $\sum_{i=0}^{m-1} (C_i(x, y) - \bar{C}(x, y))^2$ of (3) can be written as

$$\begin{aligned}
 & \sum_{i=0}^{m-1} (C_i(x, y) - \bar{C}(x, y))^2 = \\
 & = \sum_{i=0}^{m-1} C_i^2(x, y) - \frac{(\sum_{i=0}^{m-1} C_i(x, y))^2}{m}
 \end{aligned}$$

and $V_{P_{m \times N}}(x, y)$ as

$$V_{P_{m \times N}}(x, y) = \sum_{i=0}^{m-1} \sum_{j=0}^{N-1} I_{i,j}^2(x, y) - \frac{(\sum_{i=0}^{m-1} C_i(x, y))^2}{N \cdot m},$$

the new saliency map $S(x, y, m+1)$ can be computed by updating three sums at each image location:

$$\begin{aligned}
 & \sum_{i=0}^{m-1} C_i + C_m, \quad \sum_{i=0}^{m-1} C_i^2 + C_m^2, \\
 \text{and} \quad & \sum_{i=0}^{m-1} \sum_{j=0}^{N-1} I_{i,j}^2 + \sum_{j=0}^{N-1} I_{m,j}^2.
 \end{aligned}$$

Keypoints are local scale-space maxima detected in the stack of the saliency maps (3). Maxima are detected by comparing a pixel value to its 26 neighbors in a $3 \times 3 \times 3$ volume of pixels in the stack.

3 ALGORITHM IMPROVEMENTS

Five improvements are introduced in the new version of the algorithm.

3.1 Up-sampling and down-sampling

To increase the quality of the keypoint detection for small keypoint radii, the image is first up-sampled by using bi-cubic interpolation and saliency maps $S(x, y, m); m = 0, \dots, M-1$ are computed for $M = 11$. The local scale space maxima are then located for maps with $m \geq 5$. For larger keypoints, to speed up the computation, the up-sampled image is down-sampled two times. For each of the down-sampled image, the saliency maps are computed for eleven circles and the keypoint detection is applied for the maps with $m \geq 6$. Before the saliency maps are computed, the image is first smoothed with a Gaussian filter with small σ . In this way, we acquire keypoints with radii from 2.25 pixels to 21 pixels.

3.2 Computation of sums on circles

Transformation of a local region from the Cartesian to the polar representation (Fig. 1) is time consuming. Instead of addressing locations on circles to compute sums $C_m(x, y)$ and $\sum_{j=1}^N I_{m,j}^2(x, y); m = 0, \dots, M-1$, they are computed by convolving an intensity image $\mathcal{I}(x, y)$ and image $\mathcal{I}(x, y)^2$ with convolution filters

$\mathcal{A}_m; m = 0, \dots, M - 1$. We have

$$\begin{aligned} C_m(x, y) &= \mathcal{A}_m * \mathcal{I}(x, y), \\ \sum_{j=0}^{N-1} I_{m,j}^2(x, y) &= \mathcal{A}_m * \mathcal{I}^2(x, y); \\ m &= 0, \dots, M - 1. \end{aligned}$$

Here $*$ is the convolution operation in x and y and the operation $.^2$ squares the image intensity at each pixel in the image. Filter $\mathcal{A}_m(x, y)$ represents the m -th circle or 1-pixel wide annulus (see Fig. 4). The sum of weights on the circle is for each filter equal to value N . Filters $\mathcal{A}_m; m = 0, \dots, M - 1$ are generated in a double nested loop (see Algorithm 1).



Figure 4. Filters $\mathcal{A}_m(x, y); m = 1, \dots, M - 1$. The radial detector uses 11 circles to compute the saliency maps. The sum of the pixel weights is for each circle equal to N . The first filter $\mathcal{A}_0(x, y)$ with $r_m = 0$, not shown in the image, just multiplies the image intensity with N .

Result: Filters $\mathcal{A}_m; m = 1, \dots, M - 1$

$N = 720$;

for $m=1$ **to** $M-1$ **do**

$\mathcal{A}_m = \text{zeros}(2 * m + 1, 2 * m + 1)$;

$c = m + 1$;

for $n=1$ **to** N **do**

$r = m$;

$x = c + \text{round}(r * \cos(2 * \pi / N * (n - 1)))$;

$y = c + \text{round}(r * \sin(2 * \pi / N * (n - 1)))$;

$\mathcal{A}_m(x, y) = \mathcal{A}_m(x, y) + 1$;

end

end

Algorithm 1: Generation of filters $\mathcal{A}_m; m = 1, \dots, M - 1$, shown in Fig. 4.

3.3 Selecting keypoints with a large contrast

It is a common practice to apply a threshold on the saliency maps for a keypoint selection. Only the local scale-space maxima with a saliency value above the pre-specified threshold value are considered as detected keypoints. Another option is to sort the detected local scale-space maxima according to their saliency in a descending order and then take only the first n local scale-space maxima as keypoints. Usually, a larger saliency corresponds to a larger image contrast. In the case of the saliency (3), this is not true. To exclude the local extrema with a low contrast, which are more sensitive to the noise than the extrema with a large contrast, the radial detector applies a threshold on maps

$$B_{nor}(x, y, m) = \frac{\sum_{i=0}^{m-1} \alpha_i^2(x, y)}{m \cdot N \cdot I_{nor}^2} \quad (4)$$

with

$$I_{nor} = \frac{1}{3} \left(\frac{\sum_{c=1}^{N_{col}} \hat{I}_c}{N_{col}} + \frac{\sum_{r=1}^{N_{row}} \hat{I}_r}{N_{row}} + I_{max} \right).$$

Here \hat{I}_c and \hat{I}_r are the maximal intensity values of the c -th column and r -th row of the image, respectively, and I_{max} is the maximal value of the image. N_{col} and N_{row} are the number of columns and rows of the image. Hence, only those local scale-space maxima obtained on the saliency maps of (3) that have the value of (4) above the threshold value are taken into account. The same applies to the case of sorting. The local scale-space maxima are sorted according to the values of (4). Fig. 5 shows both maps $S(x, y, 5)$ and $B_{nor}(x, y, 5)$.

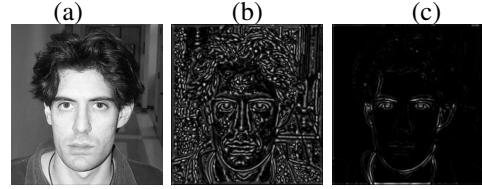


Figure 5. Saliency maps. (a) Original image, (b) Saliency map $S(x, y, 5)$ given by Eq. 3, (c) Saliency map $B_{nor}(x, y, 5)$ given by Eq. 4.

3.4 Eliminating the edge responses

The saliency measure (3) has a strong response also along the edges. These locations are unstable to small levels of the noise, therefore we would like to reject keypoints at these locations. Here we use the same technique as proposed in [9]. The two principal curvatures computed at the keypoint location and scale are considered: along the edge, the saliency will have a small principal curvature, while across the edge, the curvature is large, hence by checking for the ratio of the principal curvatures, the keypoints along the edges are discarded.

3.5 Orientation assignment

The invariance to the image plane rotation is achieved by assigning an orientation to the keypoint. Here we follow the approach proposed in [9], however we use another type of the average image for gradient computation. A radius r_m of the keypoint is used to select the average image, $\bar{\mathcal{I}}_m(x, y) = \frac{1}{mN} \sum_{j=0}^{m-1} C_j$ so that all computations are performed in a scale-invariant manner. For each image sample in the area of $3.5 \times r_m$ around the keypoint, the gradient magnitude, $g_{mag}(x, y)$, and orientation, $\theta(x, y)$, is computed using the pixel differ-

ences:

$$\begin{aligned} g_x &= \bar{\mathcal{I}}_m(x+1, y) - \bar{\mathcal{I}}_m(x-1, y) \\ g_y &= \bar{\mathcal{I}}_m(x, y+1) - \bar{\mathcal{I}}_m(x, y-1) \\ g_{mag}(x, y) &= \sqrt{g_x^2 + g_y^2} \\ \theta(x, y) &= \tan^{-1}\left(\frac{g_y}{g_x}\right). \end{aligned}$$

An orientation histogram is then formed from the gradient orientations of the sample points. The orientation histogram has 36 bins covering the 360 degree range of orientations. Each sample added to the histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a σ that is 1.5 times that of radius r_m of the keypoint. The dominant orientation is determined by detecting the global maximum in the histogram. Since the radial detector detects locations with a radial and mirror symmetry, some local maxima in the histogram can also be high, close to the value of the global maximum. Therefore, any local maximum that is within 80% of the global maximum is used to create a keypoint with that orientation. For a better accuracy, the same approach is used as in [9] to estimate the position of the maxima. A parabola is fit to three histogram values closest to the global or selected local maximum to interpolate the position of the maximum.

4 A COMPARISON OF THE DOG DETECTOR WITH THE RADIAL DETECTOR

To compare the detector based on the difference between two Gaussian filters or DoG, which is a part of the SIFT algorithm [9], with the Radial detector, we start with a two-dimensional unit height Gaussian function

$$g_K(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2 K^2}}, \quad (5)$$

which determines weighting of locations in a local region. Let us divide weights of (5) into two parts; the circular centre and its surrounding. This can be accomplished by representing the centre with the two-dimensional unit height Gaussian function with a K -times smaller standard deviation

$$g(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

and the surrounding by the difference between the two unit-height Gaussian functions

$$g_s(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2 K^2}} - e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (7)$$

Hence, $\sigma_K = K \cdot \sigma$ and $g_K(x, y) = g(x, y) + g_s(x, y)$. Fig. 6 shows the proposed decomposition. Let $G(x, y)$ and $G_K(x, y)$ denote two images obtained by convolving intensity image $I(x, y)$ with filters (6) and (5), respectively:

$$G(x, y) = g(x, y) * \mathcal{I}(x, y) \quad (8)$$

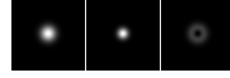


Figure 6. Decomposition of unit-height Gaussian $g_K(x, y)$ into centre $g(x, y)$ and surrounding $g_s(x, y)$. From left to right: $g_K(x, y)$, $g(x, y)$, and $g_s(x, y)$ computed for $\sigma = 3$ and $K = \sqrt{2}$.

and

$$G_K(x, y) = g_K(x, y) * \mathcal{I}(x, y). \quad (9)$$

4.1 The DoG filter and α_0 effect

Weighting functions (6) and (7) determine two zones of a circular shape (see Fig. 6). Following the idea of DEM, (8) and (9), can be used to compute two α effects but this time with local region weighting as defined by (6) and (7). Let w_K , w , and w_s be the normalizing constants of the filters $g_K(x, y)$, $g(x, y)$, and $g_s(x, y)$, respectively:

$$\begin{aligned} w_K &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g_K(x, y) dx dy = 2\pi\sigma^2 K^2, \\ w &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) dx dy = 2\pi\sigma^2, \end{aligned}$$

and

$$w_s = w_K - w = 2\pi\sigma^2(K^2 - 1).$$

Seeing $G(x, y)$ as the weighted centre and $G_K(x, y) - G(x, y)$ as the first annulus, two α_G effects can be computed by using the same idea as represented by Eq. 1:

$$\alpha_{0_G}(x, y) = \frac{G(x, y)}{w} - \frac{G_K(x, y)}{w_K}, \quad (10)$$

$$\alpha_{1_G}(x, y) = \frac{G_K(x, y) - G(x, y)}{w_s} - \frac{G_K(x, y)}{w_K}. \quad (11)$$

G denotes that the region weighting is based on the unit height Gaussian function. We can notice that the $\alpha_{0_G}(x, y)$ effect given by Eq. 10 represents convolution of intensity image $\mathcal{I}(x, y)$ by the DoG filter:

$$\alpha_{0_G}(x, y) = -\mathcal{D}(x, y) * \mathcal{I}(x, y)$$

with

$$\mathcal{D}(x, y) = \frac{K^2 g(x, y) - g_K(x, y)}{w_K}.$$

Using Eqs. 10 and 11 and $w_K = K^2 \cdot w$, the relation between effects $\alpha_{1_G}(x, y)$ and $\alpha_{0_G}(x, y)$ can easily be derived:

$$\alpha_{1_G}(x, y) = \frac{-1}{K^2 - 1} \alpha_{0_G}(x, y).$$

Hence, by knowing $\alpha_{0_G}(x, y)$, we know $\alpha_{1_G}(x, y)$ as well. For $K = \sqrt{2}$, we obtain $\alpha_{0_G}(x, y) = -\alpha_{1_G}(x, y)$. In this case, both regions, the centre and the surrounding, are equally important. Following the above, we can

conclude that the SIFT algorithm uses the $\alpha_{0_G}(x, y)$ effect as a saliency measure for the keypoint detection. SIFT uses only two α s, therefore the local scale space maxima and minima represent locations with a dark centre and light surrounding or reverse. The larger is the contrast between the centre and the surrounding the larger is the saliency. The region size is increased by increasing parameter σ . The saliency maps (3) used by the radial detector do not distinguish between the low and high contrast regions, therefore we introduce an additional measure (4). The radial detector enlarges the local region by adding a new circle to the current local region. The radial detector is slower to compute than SIFT. Computation of saliency map $S(x, y, m)$ requires filtering of two images $\mathcal{I}(x, y)$ and $\mathcal{I}^2(x, y)$. Also, filters $\mathcal{A}_m; m = 0, \dots, M - 1$ are not separable as are the Gaussian filters.

5 EXPERIMENTAL EVALUATION



Figure 7. Examples of image pairs from full-image sequences in the HPatches dataset [2]: *i_fenis*, *i_dome*, *i_leuven*, *v_there*, *v_graffiti*, and *v_sunseason*.

The radial detector is evaluated on the HPatches dataset [2] which contains 57 sequences with linear and nonlinear illumination changes and 59 sequences with viewpoint changes captured on different scenes. The HPatches dataset is built by including image sequences from several pre-existing datasets and by many new image sequences. Each image sequence consists of six images, with the first one being reference, and the remaining five containing different degrees of deformation in question. The geometric transformations with respect to the reference image are given in a form of homographies. Examples of the image pairs are shown in Fig. 7.

In experimental testing we compare the radial with the DoG detector. DoG is a part of SIFT and it is the most often used keypoint detector. In the experimental evaluation we use the protocol suggested in [17]. The number of keypoints is limited to 3000. We compute two metrics:

1) *Repeatability score*

It measures the ratio between the corresponding keypoints and the minimum number of keypoints visible in both images.

2) *Matching score*

Each keypoint is represented by the SIFT descriptor. This allows us to find matches based on the keypoint descriptor. A match is the nearest

neighbor in the descriptor space. If the match is also a geometric correspondence, then it is a correct match. The matching score is computed as the ratio between the number of correct matches and the minimum number of keypoints visible in the pair of images.

Results for the illumination and viewpoint changes are shown in Fig. 8 and Fig. 9, respectively. For both types of changes, the radial detector outperforms the DoG detector, as both tested metrics, e.g. the repeatability and matching score, are higher for a large number of image sequences. The DoG detector is usually better for the image sequences with a substantial change in the scale. The averages, computed over all image sequences, are shown in Table. 1.

6 CONCLUSIONS

The radial detector is a new version of one of the self-similarity detectors represented in [14]. The algorithm is upgraded with different techniques that speed up computation and eliminate the noise-sensitive keypoints. The keypoints are extended with a local region orientation. The experimental evaluation shows competitive results compared to the DoG detector, which is one of the most popular keypoint detectors. The radial detector is publicly available [11].

REFERENCES

- [1] P. Alcantarilla, Fernández, A. Bartoli, and A. J. Davison, KAZE features, *Proceedings of 12th European Conf. Computer Vision*, pp. 214–227, 2012.
- [2] V. Balntas, K. Lenc, A. Vedaldi, K. Mikolajczyk, Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2017.
- [3] H. Bay, A. Ess, T. Tuytelaars, L. V. Gool, Speeded-up robust features (SURF) *Comput. Vis. Image Understanding*, Vol. 110(3), pp. 346–359, 2008.
- [4] M. Brown, D. G. Lowe, Automatic panoramic image stitching using invariant features, *International Journal of Computer Vision*, Vol. 74 (1), pp. 59–73, 2007.
- [5] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J. Leonard, Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age, *IEEE Trans. Robot.*, Vol. 32 (6), pp. 1309–1332, 2016.
- [6] J. M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y. H. Jen, E. Dunn, B. Clipp, S. Lazebnik, M. Pollefeys, Building rome on a cloudless day, *Proc. of 11th European Conf. Computer Vision*, 2010, pp.368–381, 2010.
- [7] D. Ghosh, N. Kaabouch, A survey on image mosaicing techniques, *Journal of Visual Communication and Image Representation*, Vol. 34, pp. 1–11, 2016.
- [8] T. Kadir, M. Brady, Scale, Saliency and Image Description, *International Journal of Computer Vision*, Vol. 45 (2), pp. 83–105, 2001.
- [9] D. Lowe, Distinctive image features from scale invariant keypoints, *International Journal of Computer Vision*, Vol. 60 (2), pp. 91–110, 2004.
- [10] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, M. J. Milford, Visual place recognition: A survey, *IEEE Trans. Robot.*, Vol. 32 (1), pp. 1–19, 2016.

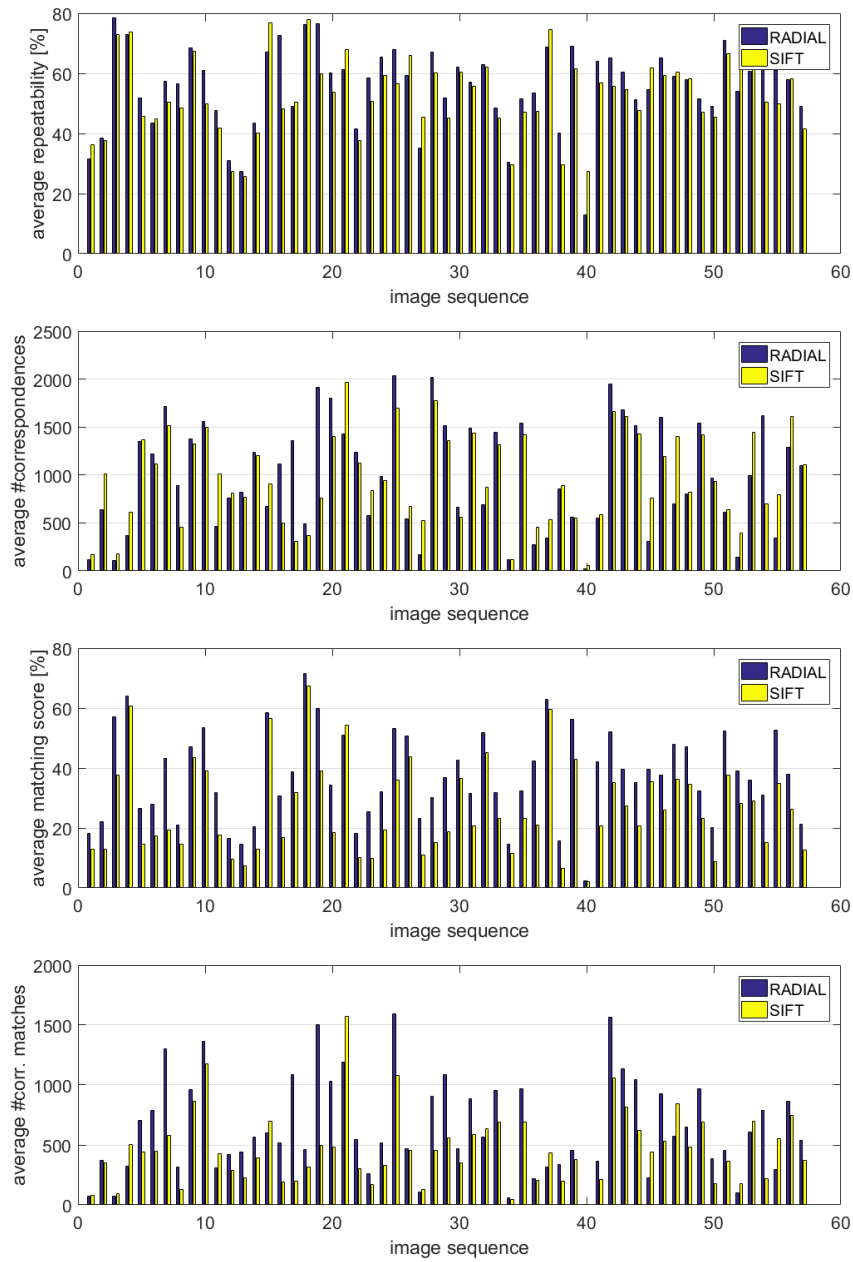


Figure 8. Illumination changes: the average values computed for repeatability scores, number of correspondences, matching scores, and number of correct matches obtained on the first 57 image sequences of the Hpatches dataset.

| Detector | Avg. Repeatability | Avg. num. Corresp. | Average Matching Score | Avg. num. Corr. Matches |
|--------------|--------------------|--------------------|------------------------|-------------------------|
| ILLUMINATION | | | | |
| Radial | 55.64 | 986 | 37.31 | 642 |
| DoG | 52.59 | 963 | 26.55 | 468 |
| VIEW | | | | |
| Radial | 49.69 | 888 | 32.87 | 731 |
| DoG | 46.62 | 916 | 26.88 | 669 |

Table 1. Average values for repeatability, number of correspondences, matching score, and correct matches computed over all image sequences with illumination and viewpoint changes of the Hpatches dataset.

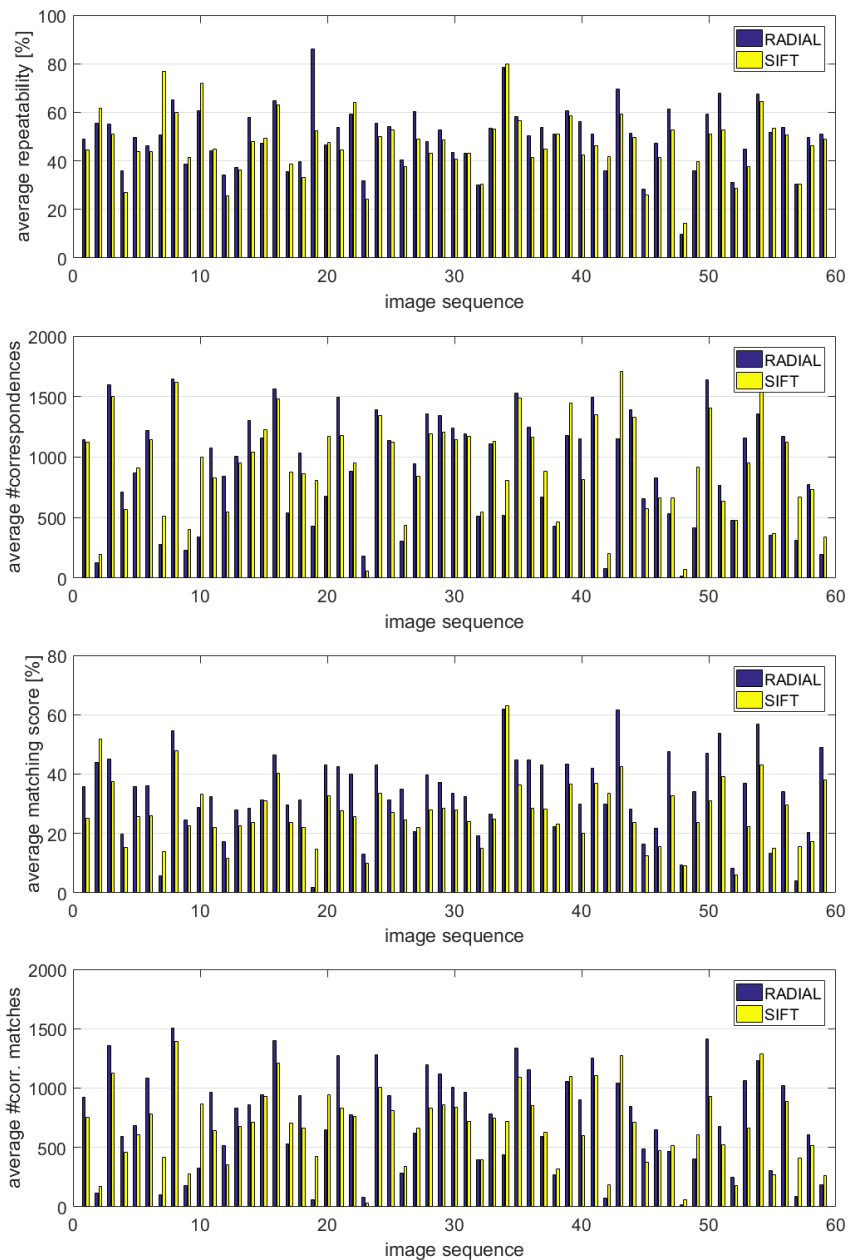


Figure 9. Viewpoint changes: the average values computed for repeatability scores, number of correspondences, matching scores, and number of correct matches obtained on the last 59 image sequences of the Hpatches dataset.

- [11] R. Mandeljc, AlphaGamma descriptor, <https://github.com/rokm/alphagamma-descriptor>, 2017.
- [12] E. Mair, G. D. Hager, D. Burschka, M. Suppa, G. Hirzinger, Adaptive and Generic Corner Detection Based on the Accelerated Segment Test, *Proc. 11th European Conf. Computer Vision*, pp. 183–196, 2010.
- [13] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust Wide Baseline Stereo from Maximally Stable Extremal Regions, *Proceedings of the British Machine Vision Conference*, pp. 36.1–36.10, 2002.
- [14] J. Maver, Self-similarity and points of interest, *IEEE Trans. Pattern Anal. Mach. Intell.* Vol. 32 (7), pp. 1211–1226, 2010.
- [15] K. Mele, D. Suc, J. Maver, Local probabilistic descriptors for image categorisation, *IET Computer Vision*, Vol. 3 (1), pp. 8–23, 2009.
- [16] K. Mikolajczyk, C. Schmid, Scale and Affine invariant interest point detectors, *International Journal of Computer Vision*, Vol. 60(1), 63–86, 2004.
- [17] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L. Van Gool, A comparison of affine region detectors. *International Journal of Computer Vision*, Vol. 65 (1/2), pp. 43–72, 2005.
- [18] E. Rosten, R. Porter, T. Drummond, Faster and Better: A Machine Learning Approach to Corner Detection, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 32 (1), pp. 105–119, 2010.
- [19] K. van de Sande, T. Gevers, C. Snoek, Evaluating color descriptors for object and scene recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 32 (9), pp. 1582–1596, 2010.
- [20] S. Smith, J. Brady, SUSAN—A new approach to low level image processing, *International Journal of Computer Vision*, Vol. 23 (1), pp. 45–48, 1997.

- [21] N. Snavely, I. Simon, M. Goesele, R. Szeliski, S. M. Seitz, Scene reconstruction and visualization from community photo collections, *Proceedings of the IEEE*, Vol. 98 (8), pp. 1370–1390, 2010.
- [22] T. Tuytelaars and K. Mikolajczyk, Local Invariant Feature Detectors: A Survey, *Foundations and Trends in Computer Graphics and Vision*, Vol. 3(3), pp. 177–280, 2008.

Jasna Maver received her PhD degree in computer science from the University of Ljubljana in 1995. From 1990 to 1991, she was with the Department of Computer and Information Science, GRASP Laboratory, University of Pennsylvania, working on the problem of next view planning. She is an Associate Professor of computer and information science at the University of Ljubljana. She holds a position at the Department of Library and Information Science and Book Studies of the Faculty of Arts. She is also a research associate of the Computer Vision Laboratory and Visual Cognitive Systems Laboratory of the Faculty of Computer and Information Science. Her research interests include methods for keypoint detection and local region representation.