Metadata of the article that will be visualized in OnlineFirst

1	Article Title	**A novel shape descriptor based on salient keypoints detection for binary image matching and retrieval**	
2	Article Sub-Title		
3	Article Copyright - Year	**Please note: Images will appear in color online but will be printed in black and white. Springer Science+Business Media, LLC, part of Springer Nature 2018 (This will be the copyright line in the final PDF)**	
4	Journal Name	Multimedia Tools and Applications	
5	Corresponding Author	Family Name	**Chatbri**
6		Particle	
7		Given Name	**Houssem**
8		Suffix	
9		Organization	Dublin City University
10		Division	Insight Centre for Data Analytics
11		Address	Dublin, Ireland
12		e-mail	houssem.chatbri@dcu.ie
13	Author	Family Name	**Kameyama**
14		Particle	
15		Given Name	**Keisuke**
16		Suffix	
17		Organization	University of Tsukuba
18		Division	Faculty of Engineering, Information and Systems
19		Address	Tsukuba, Japan
20		e-mail	keisuke.kameyama@cs.tsukuba.ac.jp
21	Author	Family Name	**Kwan**
22		Particle	
23		Given Name	**Paul**
24		Suffix	
25		Organization	University of New England
26		Division	School of Science and Technology
27		Address	Armidale, NSW, Australia
28		e-mail	paul.kwan@une.edu.au
29	Author	Family Name	**Little**
30		Particle	
31		Given Name	**Suzanne**
32		Suffix	
33		Organization	Dublin City University
34		Division	Insight Centre for Data Analytics
35		Address	Dublin, Ireland

36		e-mail	suzanne.little@dcu.ie
37	Author	Family Name	**O'Connor**
38		Particle	
39		Given Name	**Noel E.**
40		Suffix	
41		Organization	Dublin City University
42		Division	Insight Centre for Data Analytics
43		Address	Dublin, Ireland
44		e-mail	noel.oconnor@dcu.ie
45	Schedule	Received	10 July 2017
46		Revised	18 March 2018
47		Accepted	24 April 2018
48	Abstract	We introduce a shape descriptor that extracts keypoints from binary images and automatically detects the salient ones among them. The proposed descriptor operates as follows: First, the contours of the image are detected and an image transformation is used to generate background information. Next, pixels of the transformed image that have specific characteristics in their local areas are used to extract keypoints. Afterwards, the most salient keypoints are automatically detected by filtering out redundant and sensitive ones. Finally, a feature vector is calculated for each keypoint by using the distribution of contour points in its local area. The proposed descriptor is evaluated using public datasets of silhouette images, handwritten math expressions, hand-drawn diagram sketches, and noisy scanned logos. Experimental results show that the proposed descriptor compares strongly against state of the art methods, and that it is reliable when applied on challenging images such as fluctuated handwriting and noisy scanned images. Furthermore, we integrate our descriptor in a content-based document image retrieval system using sketch queries as a step for query and candidate occurrence matching, and we show that it leads to a significant boost in retrieval performances.	
49	Keywords separated by ' - '	Shape descriptors - Salient keypoints - Image matching - Sketch-based retrieval	
50	Foot note information		

Multimed Tools Appl
https://doi.org/10.1007/s11042-018-6054-x

A novel shape descriptor based on salient keypoints detection for binary image matching and retrieval

Houssem Chatbri[1] · Keisuke Kameyama[2] · Paul Kwan[3] · Suzanne Little[1] · Noel E. O'Connor[1]

Received: 10 July 2017 / Revised: 18 March 2018 / Accepted: 24 April 2018

Abstract We introduce a shape descriptor that extracts keypoints from binary images and automatically detects the salient ones among them. The proposed descriptor operates as follows: First, the contours of the image are detected and an image transformation is used to generate background information. Next, pixels of the transformed image that have specific characteristics in their local areas are used to extract keypoints. Afterwards, the most salient keypoints are automatically detected by filtering out redundant and sensitive ones. Finally, a feature vector is calculated for each keypoint by using the distribution of contour points in its local area. The proposed descriptor is evaluated using public datasets of silhouette images, handwritten math expressions, hand-drawn diagram sketches, and noisy scanned logos. Experimental results show that the proposed descriptor compares strongly against state of the art methods, and that it is reliable when applied on challenging images such as fluctuated handwriting and noisy scanned images. Furthermore, we integrate our descriptor

✉ Houssem Chatbri
houssem.chatbri@dcu.ie

Keisuke Kameyama
keisuke.kameyama@cs.tsukuba.ac.jp

Paul Kwan
paul.kwan@une.edu.au

Suzanne Little
suzanne.little@dcu.ie

Noel E. O'Connor
noel.oconnor@dcu.ie

[1] Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland

[2] Faculty of Engineering, Information and Systems, University of Tsukuba, Tsukuba, Japan

[3] School of Science and Technology, University of New England, Armidale NSW, Australia

in a content-based document image retrieval system using sketch queries as a step for query and candidate occurrence matching, and we show that it leads to a significant boost in retrieval performances.

Keywords Shape descriptors · Salient keypoints · Image matching · Sketch-based retrieval

1 Introduction

Shape matching is a vibrant area of research on image analysis and retrieval due to the numerous applications it allows [7]. Particularly, when dealing with binary images where color and texture information are absent (e.g. silhouette images, scanned documents, sketches, etc.), shape is the only available feature to be used for image representation and matching [26].

Numerous methods have been presented for shape feature extraction in binary images [54, 57]. Usually, images are subjected to contour detection or skeletonization before feature extraction in order to remove redundant information and reduce processing time [13]. Moreover, some methods select certain *keypoints* and use them to extract features [5, 35, 40]. In these cases, keypoints are selected based on their saliency or by using uniform sampling from the shape contours.

Due to the absence of background information in binary images, keypoints are extracted from the foreground pixels (i.e. regions, contours, or skeletons) and the background is omitted. In this work, we introduce a shape descriptor that approaches the problem differently by generating background information in binary images, and then involves it in feature extraction. The main steps of the descriptor are the following:

- Keypoint extraction: An image transformation is used to generate background information on the original binary image. Then, keypoints are extracted from the transformed image using pixels' local area analysis.
- Keypoint selection: An objective measure of keypoint salience is used to automatically select the most important keypoints and filter out the redundant and sensitive ones.
- Feature representation: A feature vector is calculated for each keypoint by using the distributions of contour points in the local area of the keypoint.

Our method, the binary salient keypoints (BSK) descriptor, is evaluated using silhouette images of the Kimia 216 dataset [45] and the MPEG-7 CE-shape-1 part B dataset [6], handwritten mathematical expressions of Zanibbi and Yu's dataset [56], hand-drawn diagram sketches of Liang et al.'s dataset [28], and noisy scanned logo images of the Tobacco 800 dataset [60]. Experimental results on various types of images and a comparative evaluation demonstrate that BSK is competitive compared with state of the art methods. Our code for BSK is provided online. [1]

The remainder of this paper is organized as follows: Section 2 reviews key methods of shape matching. We present our descriptor in Section 3 and evaluate it in Section 4. Concluding remarks and future work are presented in Section 5.

[1] https://github.com/hchatbri/bsk

2 Related work

Research on shape matching has led to a large repository of shape descriptors that can be classified into methods using global and local features [57], graph-based methods [28], contour-based methods and skeleton based methods [13], in addition to methods using salient keypoints [5, 32, 35, 40].

Global methods extract features using the coarse information of the shape, and hence do not convey much information about the local details. Such methods include shape signatures [43], Fourier descriptors [58], and angular partitioning [10]. Global methods are robust against noise but on the detriment of representing fine details. On the other hand, other methods take into consideration the local region of the shape points, which makes them capable of capturing fine details of the shape. Such methods include curvature scale space (CSS) [32], shape contexts [5], and variations of local binary patterns [11, 18].

Graph-based methods represent features using graph structures in contrast to statistical methods which use statistical natures of appearances [28]. Advantages of graph-based methods are their ability to represent spatial and hierarchical relationships between the object parts [8], in addition to allowing partial matching. On the other hand, graph matching requires intensive computations and thus it is common to transform a graph into a numerical feature vector in order to speed up computations, which is done at the expense of some information loss [16, 27]

Contours and skeletons have been used as an intermediate representation before feature extraction. Contours are more robust against noise than skeletons, as skeletons tend to generate noisy branches and artifacts in presence of shape border perturbations [13]. On the other hand, skeletons are more suitable in applications that require the segmentation of the original object into its constituent parts for subsequent graph-based feature representation [3, 24, 44, 51].

Some descriptors extract a number of keypoints and generate a feature vector for each one of them. Keypoints can be extracted using uniform sampling from the shape contours without special consideration about the keypoints curvature or location, offering a way to extract keypoints without a bias [29]. This has been exploited in numerous descriptors [5, 17, 40, 49], yet it does not take into account keypoints' local characteristics that make some keypoints more important than others. In addition to binary images, uniform sampled keypoints have been used on grayscale images (e.g. magnetic resonance image (MRI) registration [36]) and they have been used on color images combined with other descriptors (e.g. combined with SIFT and segmentation patches in [21] for logo retrieval).

On the other hand, keypoints that are extracted based on their salience (e.g. corners, crossing points) are biologically plausible [39], although they can lead to false detection in regions of contour perturbations or texture [46]. High curvature points of the contour have been used for keypoint extraction [20, 35]. Early methods such as Curvature scale space (CSS) uses scale space filtering [53] to extract contour inflection points [1, 32]. Then, the contour deformation and merging of inflection points caused by scale space filtering are used for feature extraction. The *CSS* method is a global statistical method, designed to deal only with closed concave contours; Convex and complex shapes are poorly represented with the technique. In [23], Kopf et al. describe an attempt to extend the *CSS* technique and make it able to represent convex shapes. Their idea is to create a mapping of the original shape to a second shape, called mapped shape, where strong convex segments of the original shape become concave segments of the mapped shape, and significant curvatures in the original shape remain significant in the mapped shape. The mapping is done by enclosing the sketch

with a circle of radius R and locate the point P of the circle closest to each sketch pixel. The sketch pixels are then mirrored on the tangent of the circle in P. The center of the circle is the average position of sketch pixels.

Scale-space filtering has also been used to extract distinctive keypoints in intensity images in the well-known SIFT descriptor [30]. However, it has been shown that SIFT keypoints are suboptimal compared to keypoints that are uniformly sampled from the shape contours when using complex binary images such as historical hieroglyphs [41]. This result is due to the absence of local changes of intensity in binary images that hinders scale-space filtering from detecting distinctive keypoints and attributing them characteristic scales. Scale-space has also been used for keypoint filtering [12], which proved to be effective but on the expense of efficiency.

Curvature information has been also used for salient keypoints extraction in [35]. Here, the salient points of a shape are defined as the higher curvature points along the shape contour that are extracted using a noise-robust approach [34]. Then, each salient point is represented with two values, the relative angular position of the salient point from the perspective of the shape centroid, and the *salience relevance* which characterizes the concavity of the contour segmented around the salient point after applying a Gaussian filter to reduce contour noise. Image matching corresponds an energy minimization function which give the distance between the best pair of corresponding salient points.

In addition to high curvature points, other salient points have been used including end points and branch points of the object's skeleton, and the vertices of the minimum enclosing rectangle of the object [59]. For each salient point, features are calculated using a circular layout of polar coordinates to calculate the distribution of some shape points which are sampled using a maximum distance method. Finally, a feature vector is constructed using a bag of words method.

Data-driven methods have been recently designed using deep learning [37, 50, 61]. Unlike the aforementioned methods, data-driven methods automatically learn salient features using convolutional layers, in an attempt to mimic the way humans perceive shapes and sketches [19]. Despite the success of such methods, they require large labeled datasets for training and they usually need graphical processing units (GPUs) to alleviate computations. Due to these reasons, engineered features remain necessary for applications where large labeled datasets are unavailable.

2.1 Our contributions

Compared to the state of the art, our descriptor's main contributions are twofold:

- We demonstrate that the background of binary images, which before has not been considered enough for feature extraction, can be used to extract distinctive features. We show that an image transform such as the distance transform (DT) can be used to enable this. Our experiments show that extracting salient keypoints using this procedure leads to improved robustness against noise that otherwise would easily corrupt object contours.
- Our descriptor is modular and it proceeds in three main steps which are feature extraction, keypoint selection and feature representation. This is similar to frameworks of widely-used color image descriptors (e.g. SIFT [30], SURF [4]). Consequently, we adapt a framework that has been used for color images into the binary image domain.

3 The proposed descriptor

The binary salient keypoints descriptor (BSK) operates as follows: First, keypoints are extracted (Section 3.1). Then, a number of keypoints are selected among the extracted ones and the others are filtered out (Section 3.2). Finally, a feature vector is calculated for each keypoint (Section 3.3).

3.1 Keypoint extraction

In this step, a transformation is applied on the input binary image in order to generate background information. Then, points having specific characteristics in their local areas are used to extract keypoints.

For our image transformation, we use the distance transform (DT) [42]. DT generates a grayscale image where the intensity of each pixel corresponds to its distance from the nearest foreground pixel (Fig. 1c). Here, the distance between pixels is equal to their Manhattan distance as commonly used in DT implementations [31].

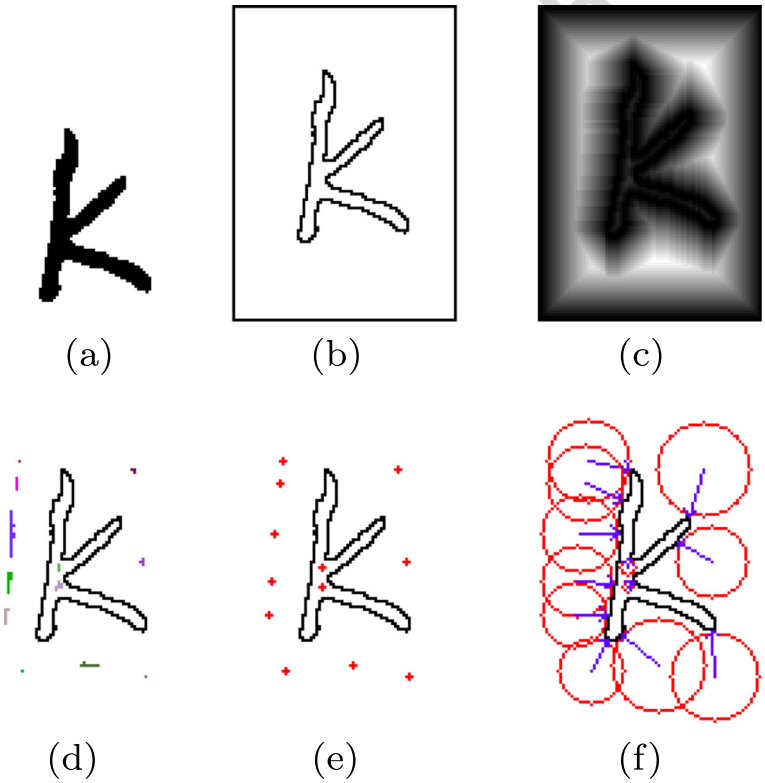


Fig. 1 Keypoint extraction steps: **a** Original binary image. **b** $W_F \times H_F$ image after normalization ($a = 0.25$). **c** DT image. **d** Regions of equal maximal intensity highlighted in different colors **e** Keypoints ($k = 11$). **f** Keypoint vectors ($\alpha = 1$): Circle radii correspond to the keypoint distance from the nearest contour point, and arrows show the orientation of the vector delimited by the keypoint and its nearest contour point

Keypoints are extracted as follows: First, the original image (Fig. 1a) is normalized by applying contour detection and image translation (Fig. 1b). Then, background information is generated using DT (Fig. 1c). Before applying DT, a 1-pixel-width border frame is added to the normalized image in order to delimit the object so DT does not systematically generate maxima at the borders. Next, regions of equal maximal intensity are detected on the DT image using a $k \times k$ square window (Fig. 1d). A region of equal maximum intensity is the contiguous pixel "islands" that have higher intensities than their neighboring pixels. They correspond to the regions of highest intensity in Fig. 1c that are shown in different colors in Fig. 1d. Finally, the detected regions are represented using their centers of masses which are taken as keypoints (Fig. 1e).

Contour detection is used to produce a compact representation of the original image that reduces the number of foreground pixels but preserves the visual information [13, 55]. Afterwards, keypoints can be extracted from regions inside and outside the object (Fig. 1e).

The dimensions (W_F, H_F) of the frame used before applying DT are calculated as follows:

$$W_F = (1 + a)\, W_{BB}, \quad H_F = (1 + a)\, H_{BB} \tag{1}$$

where W_{BB} and H_{BB} are the dimensions of the object's bounding box, and $a \geq 0$ is introduced to allow for a space between the object contours and the frame pixels in order to extract keypoints in these regions. The object is translated towards the center of the frame. In the present work, we set a empirically (Section 4.2), so that the bounding box is located in a good proximity from the foreground object (Fig. 1). A too small a would make the frame borders too close to the foreground object, which places the keypoints too close to the contour making them more vulnerable to contour noise, while a too large a would make the frame borders too far from the object contour, which increases the size of the feature extraction windows (Fig. 5) and hence puts more weight on global details of the object on the detriment of local details.

Regions of equal maximal intensity are detected using a $k \times k$ square window located at each DT image pixel. The parameter k affects the number of extracted local maxima. The larger k gets, the fewer keypoints are detected (Fig. 2). Therefore, parameter k controls the number of generated keypoints. In this paper, we set k empirically (Section 4.2.1), and we leave further investigation on setting k automatically for future work.

Due to using DT to generate background information, the extracted keypoints are in locus of symmetry between foreground pixels and thus characterize the object using its local symmetries. We anticipate the significance of such keypoints in shape representation

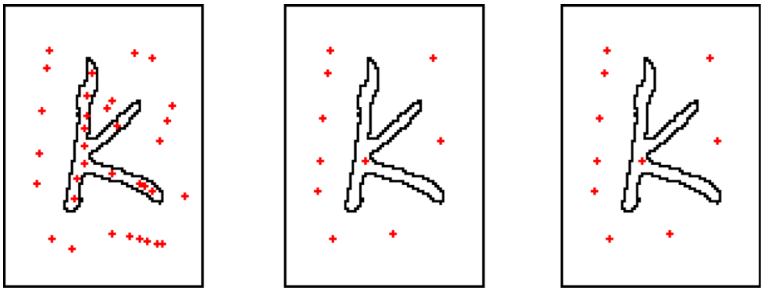


Fig. 2 Effect of the parameter k on the number of keypoints

due to the importance of symmetry as a characteristic of patterns that is exploited in human perception [52] and in computational image matching [25].

The complexity of the keypoint extraction step can be estimated as follows: The distance transform and regions of equal maximal intensity detection require two processes that browse the entire image pixels, hence they make a $2.O(n)$ complexity, with n here representing the number of image pixels. Then, keypoint detection in regions of equal maximal intensity make a complexity of $O(n)$.

3.2 Keypoint selection

The initial number of keypoints can be reduced by filtering out the redundant and sensitive keypoints. Redundant keypoints duplicate representing the same details of the image, and keypoints that are located very close to contours are sensitive to insignificant changes in image local details.

A measure of keypoint salience is introduced for keypoint ranking and selection. A salient keypoint is defined according to two aspects:

- It has few keypoints in its local area, and thus it is non-redundant.
- It is not located very close to foreground points, and thus it is robust against insignificant changes in image local details.

Formally, the salience $\gamma(i)$ of a keypoint K_i is calculated as follows:

$$\gamma(i) = \frac{d_i}{1 + n_i} \tag{2}$$

where d_i is the distance from keypoint K_i to its closest contour or frame border point, and n_i is the number of close keypoints. A keypoint K_j is considered close to K_i if it is located within a distance to K_i proportional to d_i.

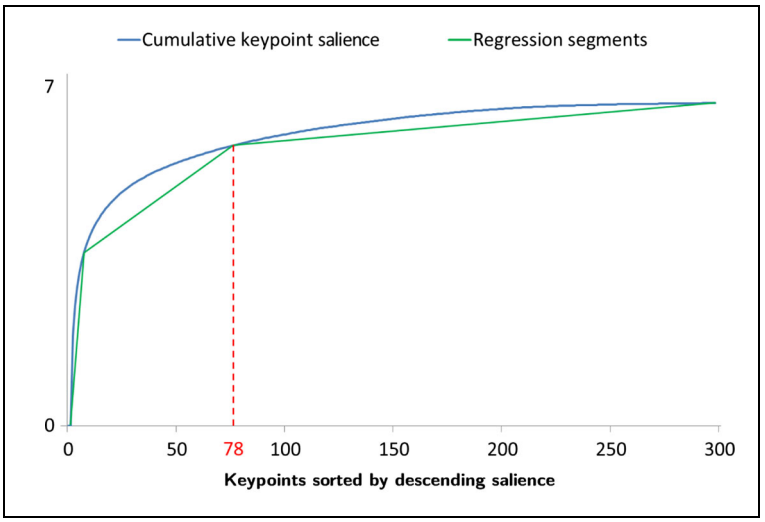
Our hypothesis for automatically selecting the most salient keypoints is as follows: We observe that the range of salience values commonly indicates three types of keypoints (Fig. 3c). The first type corresponds to few keypoints with extreme salience values, the second type corresponds to a larger number of keypoints with increasing redundancy, and the third type corresponds to keypoints with high redundancy and closeness to the contours or frame borders. Since keypoints of the third type are redundant and sensitive, they are filtered out.

In order to filter out keypoints of the third type, we calculate the cumulative keypoint salience $\Gamma(i)$ for a number i of keypoints ranked in their descending salience measures, as follows:

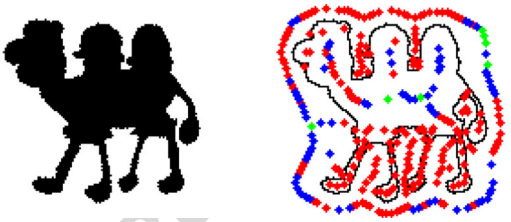
$$\Gamma(i) = ln\left(\sum_{j=1}^{i} \gamma(j)\right) \tag{3}$$

Figure 3a shows a typical curve of Γ as a function of the number of accumulated keypoints. The curve of Γ can be roughly segmented into three parts corresponding to the types of keypoints. In order to find keypoints of each type, a two-dimensional search is used to detect the three segments that minimize the area between them and the curve of Γ. Then, keypoints corresponding to the first and second types are selected. In the literature, a similar strategy has been reported in [47] to automatically detect salient corner points in online sketches using scale-space filtering and digital ink attributes (e.g. pen speed, curvature).

Figure 4 illustrates the benefit of automatic selection of keypoints using their salience scores. The top example shows matching between an image and its slightly different version

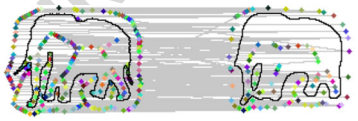


(a)

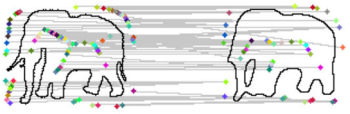


(b) (c)

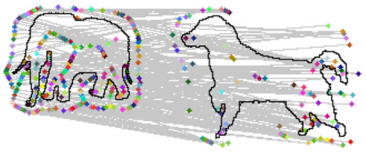
Fig. 3 Keypoint selection: **a** Curve approximation by three segments applied on image **b**, **c** keypoints of the first type in green, keypoints of second type in blue, and keypoints of third type in red. Automatic keypoint selection reduces the number of keypoints from 298 to 78



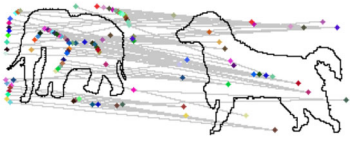
(a) $N_{Left} = 229$, $N_{Right} = 140$, $similarity = 98.95\%$



(b) $N_{Left} = 80$, $N_{Right} = 93$, $similarity = 98.95\%$



(c) $N_{Left} = 229$, $N_{Right} = 270$, $similarity = 98.30\%$



(d) $N_{Left} = 80$, $N_{Right} = 59$, $similarity = 97.76\%$

Fig. 4 Automatic keypoint selection reduces the number of keypoints while improving matching performances (*similarity* is calculated according to (7))

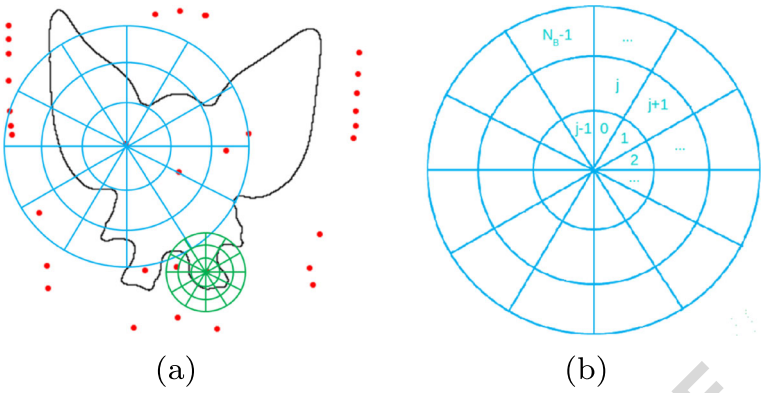


Fig. 5 Keypoint feature extraction using size-adaptive layouts

that is generated using a Gaussian filtering ($\sigma = 3$) followed by binarization [33], leading to remove the granularity of some local details. Using automatic keypoint selection does not affect the similarity between the two images, which shows that the filtered keypoints are not crucial for matching. On the other hand, the bottom example shows matching between two images belonging to different classes. Here, using automatic keypoint selection decreases the similarity, which shows that automatic keypoint selection has removed a significant number of keypoints causing false positives. In both cases, the reduction in the number of keypoints is considerable.

The complexity of the keypoint selection step can be estimated as $2.O(n^2)$, with n here representing the number of initially extracted keypoints.

3.3 Feature representation and matching

The last step is to calculate a feature vector to each keypoint K_i. For this purpose, we use a scale-invariant circular layout which radius r_i is proportional to the distance between the keypoint K_i and its closest contour point (Fig. 5):

$$r_i = \alpha \times d_i \tag{4}$$

where α is a heuristic. The idea is to set $\alpha > 1$ to allow taking into account the closest contour points in the smallest distance bins. Then, a histogram h_i is extracted by calculating the distribution of contour points in distance and angle bins, i.e. $h_i(j)$ holds the number of contour points that are located inside the feature window bin of index j (Fig. 5b). The distance between two histograms is expressed by the $\mathcal{X}^2$ statistic:

$$\mathcal{X}^2(h_1, h_2) = \frac{1}{2} \sum_{j=0}^{N_B-1} \frac{[h_1(j) - h_2(j)]^2}{h_1(j) + h_2(j)} \tag{5}$$

where N_B is the number of bins in a keypoint histogram. Using the distance d_i to set the radius of the feature layout makes the descriptor scale-invariant.

The dissimilarity d between two images I_1 and I_2 is estimated by the cumulative minimum distance between the images' keypoint histograms:

$$d(I_1, I_2) = \frac{1}{N_1} \sum_{i=0}^{N_1-1} \min_{0 \le j < N_2} \left\{ \mathcal{X}^2 \left(h_i^1, h_j^2 \right) \right\} \tag{6}$$

where N_1 and N_2 are the number of keypoints in images I_1 and I_2. Because $d(I_1, I_2)$ is asymmetric, we express the distance between two images I_1 and I_2 as follows:

$$D(I_1, I_2) = \frac{d(I_1, I_2) + d(I_2, I_1)}{2} \qquad (D \in [0, 1]) \tag{7}$$

The smaller $D(I_1, I_2)$ is, the more similar I_1 and I_2 are.

The feature vector is translation-invariant due to using the object's bounding box for image normalization. Scale-invariance is partly ensured in the keypoint filtering step (using a radius d_i of the circular region used in the salience measure (Eq. 2) that changes with the size of the image) and the feature representation step (since the radius of the feature extraction circular window depends on each keypoint and also on the object's size), but partly hindered by fixing parameter k making it scale-dependent. Rotation-invariance can be ensured by using the orientation of the vector delimited by the keypoint and its nearest contour point as a reference orientation (Fig. 1), or by using shifted matching of the keypoints' feature vectors. In case mirrored matching is necessary, it can be implemented by mirroring one of the feature vectors and repeating the matching then taking the average.

The complexity of feature representation can be estimated as $O(m).O(n)$, with m here representing the number of initially extracted keypoints, and n the number of contour points. Feature matching requires $N_B.O(m).O(n)$ with m and n are the number of keypoints in images I_1 and I_2 respectively.

4 Experiments

4.1 Datasets

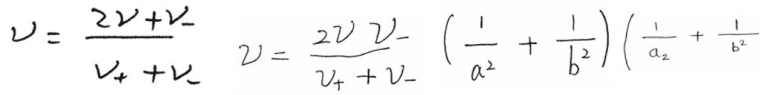
Evaluation is done using five datasets (Fig. 6): The Kimia 216 dataset [45] and the MPEG-7 dataset [6] include silhouette images that are neat and which contain single component objects. Zanibbi and Yu's dataset [56] contains handwritten mathematical expressions which exhibit handwriting fluctuations and component displacement, which also appear in Liang et al.'s dataset [28] of hand-drawn diagram sketches. The Tobacco 800 dataset [60] contains logo images that are taken from scanned documents and they are the noisiest compared to the other datasets. The datasets can be thought of as clusters' centers extracted from large datasets that are typically used in data-driven approaches [19]. On the other hand, they are fit to evaluate shape descriptors as they represent varied image classes and exhibit different challenges (e.g. noise, handwriting fluctuations).

We used Kimia, Zanibbi and Yu, and Tobacco datasets as training datasets to empirically set our parameters. The choice is made due to the reasonable sizes of these datasets, and their characteristics relative to the remaining two datasets:

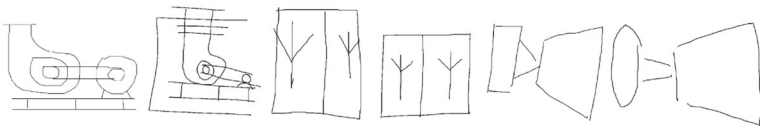
– Kimia 216 dataset can be considered a smaller subset of MPEG-7. It contains similar classes and less image variations.

(a) Kimia's dataset [41]: 216 images, 18 classes, and 12 instances

(b) MPEG-7 dataset [6]: 1400 images, 70 classes, and 20 instances



(c) Zanibbi and Yu's dataset [54]: 200 images, 20 classes, and 10 instances



(d) Liang et al. dataset [27]: 1086 images, 35 classes, and between 17 and 22 instances

(e) Tobacco 800 logo dataset [59]: 412 images, 35 classes, and between 1 and 68 instances

Fig. 6 Samples of the dataset images

– Zanibbi and Yu's dataset is handwritten, which is the same main feature of Liang et al.'s dataset.
– Tobacco dataset is used for its significant noise.

4.2 Descriptor evaluation

Before evaluating the descriptor, we set its parameters as follows: The parameter for setting the normalization frame's dimensions is set $a = 0.25$, which insures a scale-invariant frame with space between its borders and the object contours. A keypoint K_j is considered close to a keypoint K_i if the distance between them is equal or less than $\frac{d_i}{4}$, where d_i is the distance between keypoint K_i and its closest contour or frame border point. The radial and angular numbers of bins in the keypoint descriptor are set as 4 distance bins and 8 angle bins in order to make a trade-off between distinctiveness and robustness. A small

number of bins compromises the descriptor's distinctiveness, while a larger number of bins causes sensitivity to noise and fluctuations [41]. The constant for configuring the keypoint-dependent feature layout radius is set $\alpha = 1.5$ in order to insure taking into account the closest contour points in the smallest distance bins.

Evaluation is done using the *precision at n* metric [2], denoted $P@n$, which is calculated as follows:

$$P@n = \frac{|\{n\ retrieved\ images\} \cap \{relevant\ images\}|}{|\{n\ retrievd\ images\}|} \tag{8}$$

Due to variations in the number of class instances, we set the number of retrieved images n as query-dependent and equivalent to the number of the query's class instances. This makes $P@n$ equal to *precision* and *recall*. The larger $P@n$ is, the better *precision* and *recall* the descriptor shows. In the following, we specify the parameter n in $P@n$ when it is fixed (e.g. results are shown for a single dataset with a fixed number of class instances).

4.2.1 Keypoint sampling evaluation

During keypoint extraction, the parameter k defines the size of the local maxima detection window and thus affects the number of extracted keypoints (Fig. 2). We evaluate the effect of this parameter on matching performances using the Kimia 216, Zanibbi and Yu, and Tobacco datasets as training datasets for this empirical setting.

Figure 7 shows curves of $P@n$ as a function of k. We observe that the matching performance eventually decreases when k increases, and that the best matching performances correspond to $k = 3$, which means that the best way is to keep a maximum number of keypoints that will be later filtered during the keypoint selection step. According to the results of this experiment, we set $k = 3$ empirically and use it in subsequent experiments.

4.2.2 Keypoint distinctiveness evaluation

The distinctiveness of BSK's keypoints is assessed by comparison with equidistant sampling which is used in numerous descriptors, namely shape contexts [5]. We perform experiments

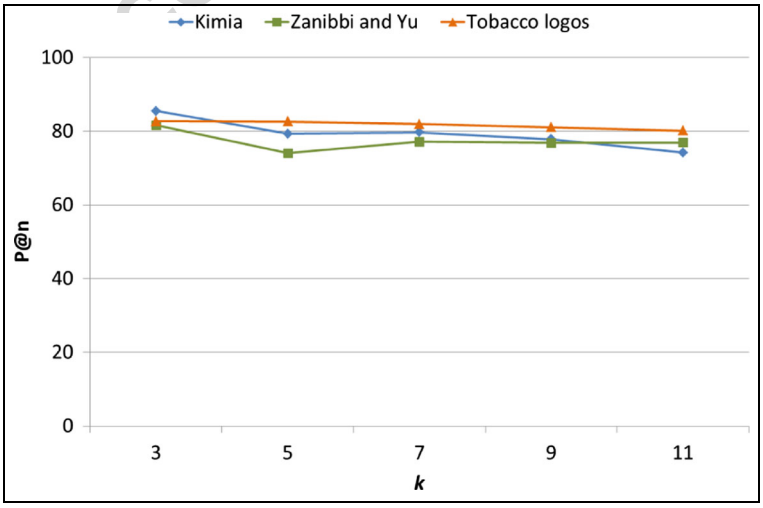


Fig. 7 Effect of varying the parameter k on $P@n$

of image retrieval using the Kimia 216 dataset where each image is used as a query and the average $P@12$ is calculated for all queries. We extract the same number of keypoints using BSK and shape contexts and perform matching using our keypoint matching steps (Section 3.3). In order to make the comparison between BSK keypoints and shape contexts fair, we introduced two modifications on the shape contexts: Features are extracted from equidistant keypoints from the contour and all the remaining contour points are considered when calculating the keypoint's histogram, unlike the original shape context descriptor where only the sampled keypoints are considered. In addition, scale-invariance is introduced by making the circular feature extraction layout's size adaptive to the shape by calculating the distance between each keypoint and its farther contour point, instead of using static log-polar layouts. Consequently, these modifications led to better results when compared with the original shape contexts considering only equidistant keypoints and using static log-polar layouts for feature extraction.

Figure 8 shows performances of BSK keypoints and shape contexts. For small numbers of extracted keypoints, using equidistant keypoints outperforms BSK keypoints. Then, starting from 40 keypoints, BSK outperforms shape contexts and the gap increases in correlation with the number of keypoints. In fact, using 40 BSK keypoints outperforms using 100 shape contexts. This result shows that our keypoints are distinctive and outperform the widely-used equidistant keypoint sampling scheme.

4.2.3 Keypoint selection evaluation

The keypoint selection step aims to reduce the number of keypoints by removing the redundant ones and the ones too close to the shape contour. Figure 9 shows retrieval performances expressed in $P@n$ as a function of the percentage of keypoints using the Kimia 216 dataset, Zanibbi and Yu's dataset, and Tobacco logos dataset. For the Kimia 216 dataset, performances increase when the percentage of keypoints increases. As for Zanibbi and Yu's and

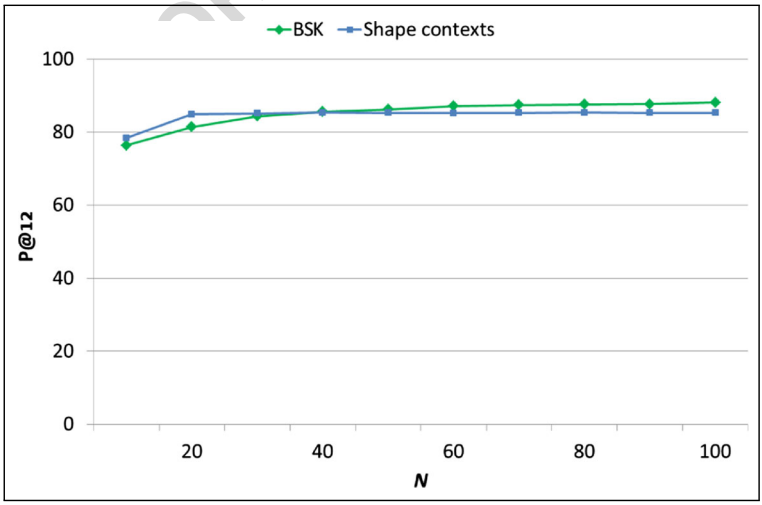


Fig. 8 $P@12$ as a function of the number of keypoints N for BSK and shape contexts on the Kimia 216 dataset

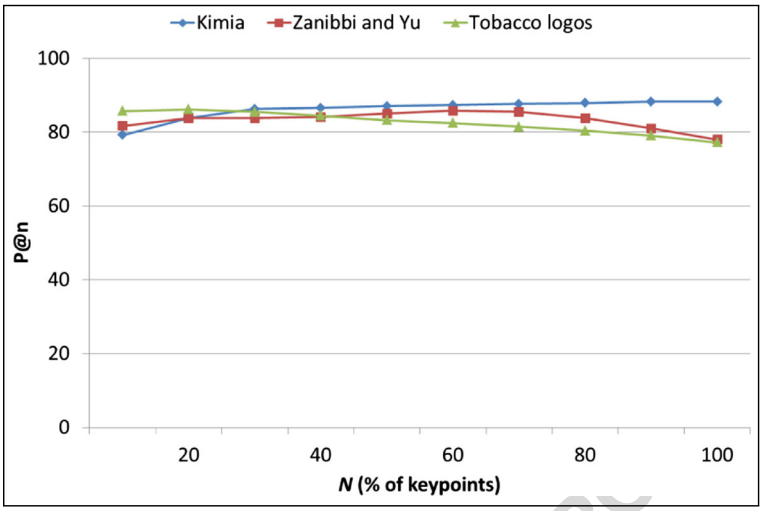


Fig. 9 $P@n$ as a function of the percentage of used keypoints relative to the total number of extracted keypoints using BSK

Tobacco logos datasets, optimal performances are obtained when not all of the keypoints are used (when 20% and 60% of keypoints are selected respectively).

Table 1 shows retrieval performances of BSK when all keypoints are used and when keypoint selection is performed. For all datasets, the reduction in number of keypoints is significant and roughly makes the third of total keypoints. In case of Zanibbi and Yu's and Tobacco logos datasets, matching performances improve. However, they decrease in case of Kimia 216 dataset. This result suggests that our keypoint salience-based selection is effective when the initial number of keypoints is relatively large (cases of Zanibbi and Yu's and Tobacco logos datasets). When the initial number of keypoints is relatively small (case of Kimia 216 dataset), the keypoint selection step would better be skipped. This can be done using a threshold on the initial number of keypoints.

4.3 Performance comparison with other descriptors

Tables 2, 3 and 4 show results of comparing our descriptor with other state of the art methods. Results are shown according to metrics that are used in available published work; In case of Kimia's dataset, we calculate the retrieval performance metric reported in several published papers, that is the number of relevant retrieved images for each of the top 6 ranks

Table 1 $P@n$ and number of keypoints N using BSK with all keypoints and with selected keypoints

Implementation	All keypoints		Selected keypoints	
	$P@n$	N	$P@n$	N
Kimia 216	88.27 %	147	85.49 %	58
Zanibbi and Yu	78.0 %	1610	81.65 %	564
Tobacco logos	77.21 %	1203	82.74 %	379

Table 2 Performances using the Kimia 216 dataset [45]

Method	BSK	SRD [11]	SC [5]	PSSG [3]
$P@6$	**93.83 %**	86.96 %	92.12 %	99.22 %

Q4

and the percentage calculated by summing these numbers. We refer to it as $P@6$. In case of the MPEG-7 dataset, the $P@20$ metric is used. In case of Liang et al.'s dataset, comparison is done using the *mean average precision (MAP)* metric [2].

BSK yielded competitive performances in all the datasets we used. Results in Tables 2–4 show that BSK is effective in case of computer generated silhouette images of the Kimia 216 dataset and the MPEG-7 dataset, and hand-drawn sketch images of Liang et al.'s dataset that exhibit high sketching perturbations and drawing style variations. In addition, BSK reached $P@10 = 82.74\%$ and $P@n = 81.65\%$ on Zanibbi and Yu's dataset [56] and the Tobacco logos dataset [60] respectively, which compares well against the support region descriptor (SRD) [11] that gave $P@10 = 47.6\%$ and $P@n = 82.55\%$. This demonstrates the effectiveness of BSK in case of handwritten images of Zanibbi and Yu's dataset and in case of the noisy scanned images of the Tobacco logos dataset. It is worth mentioning that SRD is designed to be robust against noise by combining local and global features. Results on the MPEG-7 dataset are relatively lower due to two image variations: First, the dataset has significant scale variance that challenges our descriptor, which has some scale-invariance limitations due to fixing the parameter k during the keypoint extraction step. Second, the dataset has also a significant number of mirrored images, and we do not currently take this into account during the feature vector matching (5).

Although BSK does not show supremacy over all other descriptors, results showed that it compares strongly against various types of methods. BSK outperformed shape context that uses equidistant sampling (Table 2) and other salience-based keypoint descriptors such as the contour salience descriptor (CS) [15] (Table 3), the minimal spanning tree (MST), Laplacian spectrum with geometry (LS+G) [16] and the LS+G [16] descriptors (Table 4). BSK also compares well against methods that combines local and global features such as SRD (Table 2), and graph-based methods such as MST, LS+G, and TPG [28] (Table 4). On the other hand, BSK was outperformed by PSSG [3] on the Kimia dataset [45] and TSDIZ [20] and SSD+GF [35] on the MPEG-7 dataset [6]. In case of PSSG [3], the use of skeleton pruning makes PSSG robust against contour noise, which explains the improved performances on the Kimia dataset. PSSG [3] skeletons, on the other hand, are vulnerable to shape ambiguity [38], but this problem is minor in the Kimia dataset and does not affect PSSG. As for TSDIZ [20] and SSD+GF [35], we explain the results by their invariance to scale, since SSD+GF [35] is based on the multiscale tensor scale transform, and SSD+GF [35] uses a scale-invariant salience detection that analyzes the curvature of contour points.

We further evaluate BSK's robustness against contour noise by comparing it with other keypoint-based descriptors. For this purpose, we generated noisy versions of the Kimia dataset images [45]. First, we removed the small contour perturbations using a Gaussian filter ($\sigma = 1$) followed by binarization [33]. Then, we produced 10 sets of images with

Table 3 Performances using the MPEG-7 dataset [6]

Method	BSK	CS [15]	TSDIZ [20]	SSD+GF [35]
$P@10$	**75.48 %**	36 %	81 %	85 %

Table 4 Performances using Liang et al.'s dataset [28]

Method	BSK	MST [27]	LS+G [16]	TPG [28]
MAP	**83.83 %**	29.8 %	50.9 %	61.6 %

contour noise levels from 10% to 100%. The noise is generated by a random removal of a percentage of contour pixels. Using the noisy sets of images, BSK is compared against shape contexts [5] (as used in Section 4.2.2) and a similar descriptor that uses the Harris detector [22], as a widely-used corner detector. For the three descriptors, the same number of keypoints are selected, which is equal to the number of salient keypoints selected automatically by BSK. For shape contexts, a similar number of keypoints are selected by using uniform sampling. As for the Harris-based descriptor, the keypoints are selected according to their descending Harris detector response.

Figure 10 shows examples of keypoints extracted using the three descriptors for a neat image with smooth contours and its noisy version after generating 50% contour noise. We observe that BSK and uniform sampling, used in shape contexts, produce keypoints that are sparse and cover all image details, while keypoints produced by the Harris detector tend to be localized on few corners. In order to estimate the effect of noise on keypoint location shifting, we calculate the keypoint *average shift* as follows:

$$averageshift = \frac{1}{N_2} \sum_{j=1}^{N_2} \min_i |\overrightarrow{K_i K_j}| \tag{9}$$

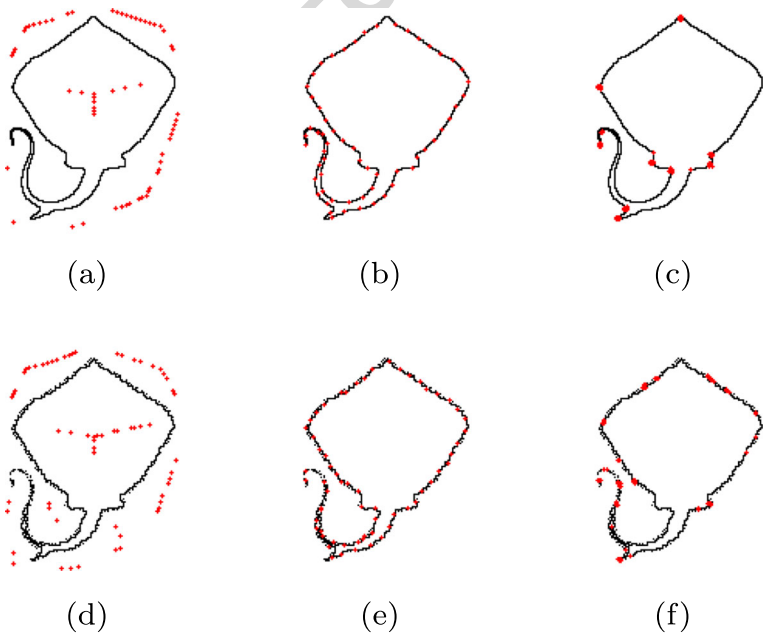


Fig. 10 Keypoints extraction methods. Top: results for a neat image (60 keypoints are extracted automatically). Bottom: results for an image with 50% contour noise (62 keypoints are extracted automatically). Right to left: BSK, equidistant sampling, and Harris detector

where N_1 and N_2 are the numbers of keypoints in the neat image and its noisy version respectively, and $\overrightarrow{K_i K_j}$ is equal to the Euclidean distance between a keypoint K_i of the neat image and a keypoint K_j in the noisy image. By taking the minimum value of $|\overrightarrow{K_i K_j}|$, we find keypoint K_i that corresponds to the previous location of keypoint K_j before a shift caused by noise occurs.

For the examples in Fig. 10, the rounded values of *average shift* for BSK, uniform sampling and Harris detector are 3 pixels, 2 pixels, and 15 pixels respectively (fixing the number of salient keypoints does not change the behavior of *average shift*). This shows that BSK and uniform sampling are more robust than Harris detector, since their keypoints do not shift much when exposed to contour noise. Accordingly, BSK and uniform sampling outperform the Harris detector in terms of retrieval performances, as shown in Fig. 11. BSK also outperforms uniform sampling although BSK's *average shift* value is slightly larger than uniform sampling's *average shift* value. This result provides an evidence that extracting keypoints from the background, instead of the contours, is a good strategy to reduce the effect of noise.

4.4 Evaluation in content-based document image retrieval

We integrate BSK in a document image retrieval system reported by Chatbri et al. [14]. This system takes input in the form of sketched mathematical expressions, and outputs a ranked list of document images that contain the user' query. This is done by using a finding the connected components of the document image that are similar to the connected components of the query using contour points distribution histograms, and then locate the ones that have a spatial arrangement similar to the query. BSK is integrated as a last step that further compares the query with the detected occurrences in the database images.

Table 5 shows a performance comparison including Chatbri et al.'s original system against when BSK is integrated, in addition to another content-based retrieval system by Zanibbi and Yu [56]. Performances are expressed with two metrics: $P - Recall$ expresses the system's ability to find the correct document image (i.e. document *page*), and $A - Recall$

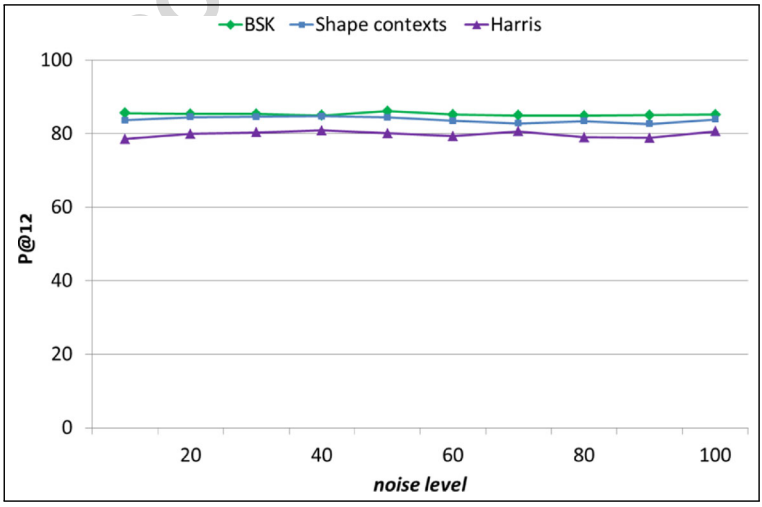


Fig. 11 $P@12$ as a function of *noise level* for BSK, shape contexts [5] and the Harris detector [22] using noisy versions of the Kimia dataset images [45]

Table 5 Average values of *P-Recall* and *A-Recall* calculated for $n = 1, 5, 10$

Method	n	Printed queries		Handwritten queries	
		P-Recall	*A-Recall*	*P-Recall*	*A-Recall*
Chatbri et al. [14]	1	100%	94.28%	40.0%	27.83%
	5	100%	96.78%	63.5%	51.15%
	10	100%	96.78%	73.5%	57.92%
Chatbri et al. [14] + BSK	1	92.5%	89.29%	54.0%	47.84%
	5	100%	96.29%	70.0%	59.89%
	10	**100%**	**96.78%**	**75.0%**	**62.43%**
Zanibbi and Yu [56]	1	.	90%	38.6%	26.7%
	5	.	90%	54.9%	39.8%
	10	.	90%	63.2%	43.3%

expresses the system's ability to find the correct *area* of the query's occurrence inside the document image [14, 56]. The metrics are calculated for the top-n retrieved document images ranked by occurrence similarity with the query.

According to the results, BSK improves retrieval performances especially when handwritten queries are used. Improvement reaches 20% of $A - Recall$ when the top-1 images are retrieved. On the other hand, performances drop in case of printed queries.

4.5 Discussion

The proposed descriptor is able to extract distinctive keypoints as demonstrated by comparison with similar numbers of shape context keypoints extracted using equidistant contour points sampling on the Kimia 216 dataset. In fact, BSK is able to outperform shape contexts using significantly fewer keypoints. This is further proven when BSK outperforms methods that detect salient points in the image contour using the other datasets. An interesting direction motivated by these results is to combine BSK keypoints with salient keypoints of the contour for the sake of better distinctiveness.

Experiments on challenging images, such as fluctuated handwritten mathematical expressions of Zanibbi and Yu's dataset and hand-drawn diagram sketches of Liang et al.'s dataset, demonstrate the reliability of BSK, as it outperforms largely other methods. Methods used for comparison include graph-based descriptors which are known for their high matching performances and ability to perform partial matching. The reliability of BSK is further demonstrated when assessed on the noisy scanned images of the Tobacco logos dataset.

The keypoint selection based on keypoint salience is effective in reducing the number of keypoints without significantly compromising the descriptor's distinctiveness. However, the performances improve when the initial number of keypoints is relatively large. For this purpose, a threshold on the initial number of keypoints can be used to activate or skip the salience-based keypoint selection.

BSK is adequate to be used for applications of image retrieval from document image databases. This is shown by the performance improvement it leads to when integrated in a standard document image retrieval system.

Finally, BSK is currently not suitable for real-time applications. In order to become so, the following can be done:

– Use parallel computing: Currently, all the procedures are executed sequentially due to limited memory. Otherwise, several steps of the algorithm can be made faster, including a parallel implementation of the distance transform [9], faster connected components extraction, and feature extraction for each keypoint in parallel.
– Resize the images to a reasonably smaller scale, which will speed up the steps aforementioned, and use integral images [48] to speed up local processes.

5 Conclusions and future work

In this paper, we introduced a descriptor for binary image matching using image salient keypoints. The proposed binary salient keypoints descriptor (BSK) generates background information in binary images, then extracts keypoints using pixels that have specific characteristics in their local areas. A measure of keypoint salience is used for automatically selecting the most salient keypoints and filtering out the redundant and sensitive ones.

The proposed descriptor has been evaluated using five public datasets of silhouette images, handwritten mathematical expressions, hand-drawn diagram sketches, and scanned logo images. Experimental results and comparison with state of the art methods demonstrated that BSK has competitive matching performances when applied on various types of images, including challenging images of fluctuated handwriting and noisy scanned images. Furthermore, BSK's integration in a content-based document image retrieval system leads to improving the system's performances considerably.

BSK paves the way for future research on salient keypoints detection in the background of binary images, as an unconventional new way of binary image analysis. In addition, it can be improved by tuning its keypoint extraction, filtering, and feature representation modular stages. We identify areas of future work as follows:

– Scale-invariance can be improved by setting parameter k automatically. For instance, instead of fixing the value of k according to empirical results on a number of datasets, k can be set according to each image taking into account its characteristics that can lead to produce more keypoints (e.g. scale, texture). On the other hand, the feature vector matching equation (5) can be modified to implement mirror matching.
– On the other hand, it would be interesting to make the parameters of BSK set in an evolutionary or data-driven way. For instance, one can try using a genetic algorithm where the genetic representation uses BSK's parameters and the fitness function is a performance metric (e.g. P@n) in a training dataset.
– We introduce a specific definition of keypoint salience that is based on the proximity between keypoints and the distance between a keypoint and the object contour. Alternatively, other definition of keypoint salience can be defined for specific applications and compared.

In addition to image matching, it would be interesting to investigate applying our descriptor in similar applications such as image registration, particularly magnetic resonance (MR) images where the shape information provides significant features [36]. Moreover, it would be interesting to apply our descriptor in color images, in combination with other descriptors. For instance, BSK would be fit to embed in the logo retrieval framework proposed in [21], preceded by edge detection, and combined with color images features (SIFT and segmentation patches).

Acknowledgments This work has emanated from a research grant in part from the Monbukagakusho Scholarship sponsored by the Japanese Government, in part from the Irish Research Council (IRC) under Grant Number GOIPD/2016/61, and in part from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 (Insight Centre for Data Analytics). The authors would also like to thank Dr. Richard Zanibbi for providing his dataset [56], Dr. Mathieu Delalandre and Dr. Alireza Alaei for their assistance with extracting logos from the Tobacco 800 dataset, and Dr. Shuang Liang for her assistance with her dataset [28].

References

1. Abbasi S, Mokhtarian F, Kittler J (1999) Curvature scale space image in shape similarity retrieval. Multimed Syst 7(6):467–476
2. Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval, vol 463. ACM Press, New York
3. Bai X, Latecki LJ (2008) Path similarity skeleton graph matching. IEEE Trans Pattern Anal Mach Intell 30(7):1282–1292
4. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: European conference on computer vision (ECCV), pp 404–417
5. Belongie S, Malik J, Puzicha J (2002) Shape matching and object recognition using shape contexts. IEEE Trans Pattern Anal Mach Intell 24(4):509–522
6. Bober M (2001) MPEG-7 visual shape descriptors. IEEE Trans Circ Syst Vid Technol 11(6):716–719
7. Breuß M (2013) Innovations for shape analysis: models and algorithms. Springer Science & Business Media
8. Bunke H, Riesen K (2012) Towards the unification of structural and statistical pattern recognition. Pattern Recogn Lett 33(7):811–825
9. Cao T-T, Tang K, Mohamed A, Tan T-S (2010) Parallel banding algorithm to compute exact distance transform with the gpu. In: ACM SIGGRAPH Symposium on interactive 3d graphics and games. ACM, pp 83–90
10. Chalechale A, Naghdy G, Mertins A (2005) Sketch-based image matching using angular partitioning. In: IEEE Transactions on systems, man and cybernetics, part A: systems and humans
11. Chatbri H, Kameyama K, Kwan P (2013) Sketch-based image retrieval by size-adaptive and noise-robust feature description. In: International conference on digital image computing: techniques and applications (DICTA). IEEE, pp 1–8
12. Chatbri H, Davila K, Kameyama K, Zanibbi R (2015) Shape matching using keypoints extracted from both the foreground and the background of binary images. In: International Conference on image processing theory, tools and applications (IPTA). IEEE, pp 205–210
13. Chatbri H, Kameyama K, Kwan P (2015) A comparative study using contours and skeletons as shape representations for binary image matching. Pattern Recognition Letters
14. Chatbri H, Kameyama K, Kwan P (2015) Towards a segmentation and recognition-free approach for content-based document image retrieval of handwritten queries. In: Asian Conference on pattern recognition (ACPR). IAPR
15. da S Torres R, Falcao AX (2007) Contour salience descriptors for effective image retrieval and analysis. Image Vis Comput 25(1):3–13
16. Demirci FM, van Leuken RH, Veltkamp RC (2008) Indexing through laplacian spectra. Comput Vis Image Underst 110(3):312–325
17. Donoser M, Riemenschneider H, Bischof H (2010) Efficient partial shape matching of outer contours, 281–292
18. Dubey SR et al (2016) Multichannel decoded local binary patterns for content-based image retrieval. IEEE Trans Image Process 25(9):4018–4032
19. Eitz M, Hays J, Alexa M (2012) How do humans sketch objects? ACM Trans Graph 31(4):44–1
20. Fernanda AA, Paulo AV, da S Torres MR, Falcão AX (2010) Shape feature extraction and description based on tensor scale. Pattern Recogn 43(1):26–36
21. Fu J, Wang J, Lu H (2010) Effective logo retrieval with adaptive local feature selection. In: ACM International conference on multimedia (ACM MM). ACM, pp 971–974
22. Harris C, Stephens M (1988) A combined corner and edge detector. In: Alvey vision conference, pp 147–151
23. Kopf S, Haenselmann T, Effelsberg W (2005) Enhancing curvature scale space features for robust shape classification. In: International conference on multimedia and expo (ICME). IEEE, p 4–pp

24. Laiche N, Larabi S, Ladraa F, Khadraoui A (2014) Curve normalization for shape retrieval. Signal Process Image Commun 29(4):556–571
25. Lee S (2013) Symmetry-driven shape description for image retrieval. Image Vis Comput 31(4):357–363
26. Liang S, Sun Z (2008) Sketch retrieval and relevance feedback with biased svm classification. Pattern Recogn Lett 29(12):1733–1741
27. Liang S, Li R-H, Baciu G (2011) A graph modeling and matching method for sketch-based garment panel design. In: International conference on cognitive informatics & cognitive Computing (ICCI CC). IEEE, pp 340–347
28. Liang S, Luo J, Wenyin L, Wei Y (2015) Sketch matching on topology product graph. IEEE Trans Pattern Anal Mach Intell 37(8):1723–1729
29. Ling H, Jacobs DW (2007) Shape classification using the inner-distance. IEEE Trans Pattern Anal Mach Intell 29(2):286–299
30. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis
31. Meijster A, Roerdink JBTM, Hesselink WH (2000) A general algorithm for computing distance transforms in linear time. In: Mathematical morphology and its applications to image and signal processing. Springer, pp 331–340
32. Mokhtarian F, Abbasi S, Kittler J (1996) Robust and efficient shape indexing through curvature scale space. In: British machine and vision conference (BMVC), vol 96
33. Otsu N (1975) A threshold selection method from gray-level histograms. Automatica 11(285-296):23–27
34. Pedrosa GV, Barcelos CAZ (2010) Anisotropic diffusion for effective shape corner point detection. Pattern Recogn Lett 31(12):1658–1664
35. Pedrosa GV, Batista MA, Barcelos CAZ (2013) Image feature descriptor based on shape salience points. Neurocomputing
36. Qi W, Zou C, Yuan Y, Hongbing L, Yan P (2013) Image registration by normalized mapping. Neurocomputing 101:181–189
37. Qian Y, Yang Y, Song Y-Z, Xiang T, Hospedales T (2015) Sketch-a-net that beats humans arXiv:1501.07873
38. Ren Z, Yuan J, Meng J, Zhang Z (2013) Robust part-based hand gesture recognition using kinect sensor. IEEE Trans Multimed 15(5):1110–1120
39. Richards W, Hoffman DD (1985) Codon constraints on closed 2D shapes. Comput Vis Graph Image Process 31(3):265–281
40. Roman-Rangel E, Marchand-Maillet S (2014) Hoosc128: a more robust local shape descriptor. In: Pattern recognition, volume 8495 of Lecture notes in computer science. Springer, pp 172–181
41. Roman-Rangel E, Marchand-Maillet S (2015) Shape-based detection of maya hieroglyphs using weighted bag representations. Pattern Recogn 48(4):1161–1173
42. Rosenfeld A, Pfaltz JL (1966) Sequential operations in digital picture processing. J ACM (JACM) 13(4):471–494
43. Roy Davies E (2004) Machine vision: theory, algorithms, practicalities. Elsevier
44. Saha PK, Borgefors G, Sanniti di Baja G (2015) A survey on skeletonization algorithms and their applications. Pattern Recognition Letters
45. Sebastian T, Klein P, Kimia B (2001) Recognition of shapes by editing shock graphs. In: International conference on computer vision (ICCV), vol 1. IEEE, pp 755–755
46. Sebe N, Qi T, Loupias E, Lew MS, Huang TS (2003) Evaluation of salient point techniques. Image Vis Comput 21(13):1087–1095
47. Sezgin TM, Davis R (2007) Scale-space based feature point detection for digital ink. In: ACM SIGGRAPH 2007 courses. ACM, p 36
48. Shafait F, Keysers D, Breuel TM (2008) Efficient implementation of local adaptive thresholding techniques using integral images. In: Document recognition and retrieval XV
49. Shu X, Xiao-Jun W (2011) A novel contour descriptor for 2D shape matching and its application to image retrieval. Image Vis Comput 29(4):286–294
50. Song J, Song Y-Z, Xiang T, Hospedales T, Ruan X (2016) Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval. In: British Machine vision conference (BMVC), vol 3
51. Sundar H, Silver D, Gagvani N, Dickinson S (2003) Skeleton based shape matching and retrieval. In: Shape modeling international. IEEE, pp 130–139
52. Tyler CW (2002) Human symmetry perception and its computational analysis. Psychology Press
53. Witkin AP (1984) Scale-space filtering: a new approach to multi-scale description. In: International conference on acoustics, speech, and signal processing (ICASSP), vol 9. IEEE, pp 150–153
54. Yang M, Kpalma K, Ronsin J (2008) A survey of shape feature extraction techniques. Pattern Recogn, 43–90

631 55. Yang X, Liu H, Latecki LJ (2012) Contour-based object detection as dominant set computation. Pattern
632 Recogn 45(5):1927–1936
633 56. Zanibbi R, Yu L (2011) Math spotting: retrieving math in technical documents using handwritten query
634 images. In: International Conference on document analysis and recognition (ICDAR)
635 57. Zhang D, Guojun L (2004) Review of shape representation and description techniques. Pattern Recogn
636 37(1):1–19
637 58. Zhang D, Lu G (2002) A comparative study of fourier descriptors for shape representation and retrieval.
638 In: Asian Conferernce on computer vision (ACCV), pp 646–651
639 59. Zhao Peng, Guoqin Wu, Yijuan Lu, Xianwen Wu, Yao Sheng (2016) A novel hand-drawn sketch
640 descriptor based on the fusion of multiple features. Neurocomputing 213:66–74
641 60. Zhu G, Doermann D (2007) Automatic document logo detection. In: International conference on
642 document analysis and recognition (ICDAR), pp 864–868
643 61. Zhu F, Xie J, Fang Y (2016) Learning cross-domain neural networks for sketch-based 3d shape retrieval.
644 In: AAAI Conference on artificial intelligence. AAAI Press, pp 3683–3689

Houssem Chatbri is a Postdoctoral Researcher at the Insight Centre for Data Analytics, Dublin City University in Ireland. He graduated from University of Tsukuba in Japan in 2016. His research interests include multimedia retrieval, image and video analysis and computer vision.

Keisuke Kameyama is a Professor of computer science at the Faculty of Engineering, Information and Systems, University of Tsukuba in Japan. His research interest include pattern recognition, machine learning, signal processing, computational intelligence and multimedia retrieval. He graduated from Tokyo Institute of Technology in 1991.

Paul Kwan is a Professor of computer science at the University of New England in Australia. He received a BSc and an MSc degree in computer science from Cornell University and University of Arizona (USA) in 1986 and 1988, respectively, then a PhD degree in 2003 from University of Tsukuba, Japan. His research interests include artificial intelligence, computer vision and image processing, complex systems and computational modelling, and bioinformatics.

Suzanne Little is a Lecturer at the School of Computing, Dublin City University, Ireland. She completed her PhD at the University of Queensland, Australia. Her research interests include machine learning, computer vision, multimedia analytics, semantic search and data integration in various applications such as security, technology enhanced learning, biomedical, multimedia archives (news), autonomous vehicles, internet of things and smart communities.

Noel E. O'Connor is a Professor in the School of Electronic Engineering, the Director of Information Technology and the Digital Society Hub, and a Principal Investigator at the Insight Centre for Data Analysis, Dublin City University (DCU), Ireland. obtained his PhD from Dublin City University in 1998. His research interests include machine learning, image and video analysis and computer vision with various applications such as autonomous vehicles, e-learning smart cities an security.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES:

Q1. Please check affiliations if captured and presented correctly.
Q2. Houssem Chatbri was captured as the corresponding author. Please check if correct.
Q3. Please check section heads if assigned to its appropriate levels.
Q4. Please provide significance of bold emphasis of tables 2–5.