

Analysis of the Multi-Dimensional Scale Saliency Algorithm and Its Application to Texture Categorization

Pablo Suau and Francisco Escolano

Robot Vision Group, University of Alicante, Spain
{pablo,sco}@dccia.ua.es

Abstract. A new approach for multi-dimensional Scale Saliency (MDSS) was lately introduced. In this approach, the Scale Saliency algorithm by Kadir and Brady is extended to the multi-dimensional domain. The MDSS algorithm is based on alternative entropy and divergence estimation methods whose complexity does not increase exponentially with data dimensionality. However, MDSS has not been applied to any practical problem yet. In this paper we apply the MDSS algorithm to the texture categorization problem, and we provide further experiments in order to assess the suitability of different estimators to the algorithm. We also propose a new divergence measure based on the k-d partition algorithm.

1 Introduction

High level vision tasks usually rely on the results provided by image processing or feature extraction algorithms. The interest regions detected by feature extraction methods should satisfy several properties: they must be informative, distinguishable and invariant to a wide range of transformations¹. The work in this paper is focused on the Scale Saliency algorithm by Kadir and Brady [1]. This algorithm is theoretically sound, due to the fact that it uses Information Theory in order to search the most informative regions on the image. Although its poor performance for matching problems [2], it has been shown to perform well in image categorization tasks [3]. Furthermore, it has been successfully applied before to this kind of problems [4,5].

The Scale Saliency algorithm [1] detects salient or unpredictable regions on an image. Shannon's entropy is used to measure the saliency of an image region. Given a pixel x , its entropy at scale s is computed from the grayscale intensity pdf of the circular region R_x of radius s , centered over x . The intensity pdf is approximated by means of an intensity histogram where $P_{d,s,x}$ is the probability that the intensity value $d \in D$ is found in R_x (in the case of a grayscale image, $D = \{0, \dots, 255\}$).

¹ Several authors prefer the term *covariant*, referring to image features that adapt to the transformation applied to the image.

The algorithm works as follows: firstly, entropy is estimated for all pixels x in the image, using all scales s in a range of scales between s_{min} and s_{max} (Eq. 1). Next, entropy peaks (local maxima in scale space) are selected (Eq. 2). Then, entropy peaks are weighted by means of a self-dissimilarity metric between scales (Eq. 3). Finally, a subset of the salient features is selected, in order of weighted entropy (Eq. 4). These selected features are the most salient features of the image.

$$H(s, x) = \sum_{d \in D} P_{d,s,x} \log_2 P_{d,s,x} \tag{1}$$

$$S = \{s : H(s - 1, x) < H(s, x) > H(s + 1, x)\} \tag{2}$$

$$W(s, x) = \frac{s^2}{2s - 1} \sum_{d \in D} |P_{d,s,x} - P_{d,s-1,x}| \tag{3}$$

$$Y(s, x) = H(s, x)W(s, x) . \tag{4}$$

The application of the algorithm summarized above to higher dimensional data is straightforward. For instance, in RGB color images, where each pixel is assigned three different intensity values (corresponding to the three RGB channels), the local intensity pdf may be estimated from a 3D histogram. In general, for n D data, the same algorithm can be applied if entropy and self-dissimilarity are computed from n D histograms. Two problems arise from this extension to the multi-dimensional domain, due to the curse of dimensionality. Firstly, the complexity order of the algorithm increases exponentially with data dimensionality. And secondly, higher dimensional data yields sparser histograms, that are less informative. These issues make the use of the original Scale Saliency algorithm unfeasible in the case of $n \geq 4$ dimensions.

We previously introduced two extensions of the Scale Saliency algorithm to the multi-dimensional domain, based on entropy and self-dissimilarity (divergence between scales) estimation from entropic graphs [6] and from k-d partitions [7]. Our experiments show that up to 31 dimensions can be processed with MDSS, but i) apart from a repeatability test, we do not provide additional evidence of the suitability of the applied estimators to the Scale Saliency task, ii) the theoretical background of our k-d partition based divergence is not discussed, and iii) no practical application of the MDSS is reported. In this paper we address these three points. Firstly, in Sects. 2 and 3 we summarize the two different MDSS approaches (entropic graphs and k-d partition based), introducing a new k-d partition divergence estimation method. Then, in Sect. 4 we assess these approaches. Finally, in Sect. 5, we apply the MDSS algorithm to the texture categorization problem.

2 MDSS Based on k-Nearest Neighbour Graphs

In this approach, each pixel $x_i \in X$ is represented as a d -dimensional vector. The neighbourhood R_x of a pixel is represented by an undirected and fully connected graph $G = (V, E)$, being the nodes $v_i \in V$ the d -dimensional vectors representing

$x_i \in R_x$ and E the set of edges connecting each pair of nodes. The weight of each edge is the Euclidean distance in \mathcal{R}^d between its two incident nodes. Entropy and divergence are estimated from the K-Nearest Neighbour Graph (KNNG), a subset of the fully connected graph, that connects each node to its k neighbours. From the KNNG, entropy is estimated by means of the measure defined by Kozachenko and Leonenko [8]

$$\hat{H}_{N,k} = \frac{1}{N} \sum_{i=1}^N \log \left((N-1) e^{-\psi(k)} B_d(\rho_{k,N-1}^{(i)})^d \right) , \tag{5}$$

where $|V| = N$, B_d is the volume of the d -dimensional unit ball, $\rho_{k,N-1}^{(i)}$ is the distance to the k -nearest neighbour of i when taking the rest of $N-1$ samples, and $\psi(z)$ is the digamma function.

In the case of self-dissimilarity between scales, the Friedman-Rafsky test is applied [6]. Let s be the scale in which an entropy peak was found. In order to weight that entropy value, we must calculate the dissimilarity with respect to scale $s-1$. Let X_s and X_{s-1} be the set of nodes of R_x at scales s and $s-1$, respectively. Since $X_{s-1} \subset X_s$ (new pixels are added to the previous ones as we increase the scale), the test only requires to build the KNNG from X_s and to count the amount of edges in this KNNG that connect a node from X_s/X_{s-1} to a node from X_{s-1} . One minus this number of edges is a consistent estimator of the Henze and Penrose divergence.

3 MDSS Based on the k-d Partition Algorithm

The second MDSS approach is based on the k-d partition algorithm by Stowell *et al.* [9]. As in the approach presented above, each pixel in R_x is represented as a d -dimensional vector. The d -dimensional feature space is recursively spit into cells following the data splitting method of the k-d tree algorithm. At each level, the data is spit by their sample median along one axis. Then, data splitting is applied to each subspace until an uniformity stop criterion is reached. The aim of this stop criterion is to produce cells with uniform empirical distribution, in order to best approximate the underlying pdf. The data partition yields a set $A = \{A_j\}$ of p cells, and then entropy estimation is given by

$$\hat{H} = \sum_{j=1}^p \frac{n_j}{n} \log \left(\frac{n}{n_j} \mu(A_j) \right) , \tag{6}$$

where $\mu(A_j)$ is the volume of the cell A_j , n_j is the number of samples in A_j and n is the the total number of samples in R_x .

Regarding the self-dissimilarity between scales, we propose a new divergence metric inspired by the k-d partition algorithm. Our k-d partition based divergence metric follows the spirit of the total variation distance [10], but may also be interpreted as a L1-norm distance. The total variation distance between two probability measures P and Q in the case of a finite alphabet is given by

$$\delta(P, Q) = \frac{1}{2} \sum_x |P(x) - Q(x)| . \quad (7)$$

Let $f(x)$ and $g(x)$ be two distributions, from which we gather a set X of n_x samples and a set O of n_o samples, respectively. If we apply the partition scheme of the k-d partition algorithm to the set of samples $X \cup O$, the result is a partition A of $X \cup O$, being $A = \{A_j | j = 1, \dots, p\}$. In the case of $f(x)$, the probability of any cell A_j is given by $p(A_j) = \frac{n_{x,j}}{n_x} = p_j$ where $n_{x,j}$ is the number of samples from X in cell A_j . Conversely, in the case of $g(x)$ the probability of each cell A_j is given by $q(A_j) = \frac{n_{o,j}}{n_o} = q_j$ where $n_{o,j}$ is the number of samples from X in the cell A_j . Since both sample sets share the same partition A , and considering the set of cells A_j a finite alphabet, we can compute the total variation distance between $f(x)$ and $g(x)$ as

$$D(O||X) = \frac{1}{2} \sum_{j=1}^p |p_j - q_j| . \quad (8)$$

The latter distance metric can be used as a self-dissimilarity measure in Scale Saliency algorithm, since it satisfies $0 \leq D(O||X) \leq 1$. The minimum value $D(O||X) = 0$ is obtained when all the cells A_j contain the same proportion of samples from X and O . By the other hand, the maximum value $D(O||X) = 1$ is obtained when all the samples in any cell A_j were gathered from a single distribution.

4 Experimental Results

In this section we introduce additional experiments to those shown in [7]. These experiments in [7] were aimed to compare the computational time of both MDSS approaches and the quality of the extracted features. We demonstrated that the computational order decreased from exponential with respect to data dimensionality (due to the use of histograms in the original Kadir and Brady algorithm) to linear. The computational efficiency of the k-d partition approach is remarkably higher when compared to the rest of algorithms; it can process a 31-dimensional 256×256 image in less than four minutes. In the case of the quality of the extracted features, we applied a repeatability test in order to assess the stability of the extracted features over a wide range of transformations, using the image dataset proposed by Mikolajczyk *et al.* [2]. Colour information was used not only in the case of MDSS, but also in the case of the Kadir and Brady Scale Saliency. The results showed that none of the MDSS approaches performs better than the other one in all circumstances. Furthermore, both MDSS algorithms showed lower repeatability than the original Scale Saliency algorithm.

4.1 Entropy Estimation Bias

Firstly we assess the estimation bias of the two entropy estimation methods summarized above, using two types of distributions: Gaussian and uniform. The

normal distribution $N(\mu, \sigma^2)$ has maximum entropy among all real-valued distributions with specified mean μ and standard deviation σ [11]. By the other hand, the uniform distribution on the interval $[a, b]$ is the maximum entropy distribution among all continuous distributions which are supported in the interval $[a, b]$ [11]. In both cases we measured the mean deviation (after 100 runs) from the theoretical entropy of the Gaussian and uniform distributions for increasing data dimensionality and a number of samples corresponding to the number of pixels taken by the MDSS algorithm between scales $s_{min} = 3$ and $s_{max} = 30$. The results are shown in Fig. 1. As one may expect, in general the estimation asymptotically improves when increasing the number of samples. Also, in all cases, increasing data dimensionality degrades the entropy estimation. None of the tested estimators performs better in all circumstances. The Leonenko *et al.* estimator approximates better the theoretical entropy of the Gaussian distribution, while the k-d partition estimation approximates better the theoretical entropy of the uniform distribution. It must be also noted that the Leonenko estimator does not require a high value of the parameter k ; on the contrary, it yields better results for $k = 2$.

Despite these results, the Scale Saliency algorithm does not require an exact estimation of entropy, as long as the saliency estimator used follows the trend of Shannon’s entropy as saliency increases. We performed an additional experiment in order to test the trend of the entropy estimation given by the k-d partition algorithm and the Leonenko *et al.* estimator. The experiment consisted in gathering N samples $x \in [0, 255]^d$ from a Gaussian and an uniform distribution, being N the number of pixels processed at $s_{max} = 30$ during the Scale Saliency algorithm. Then we computed the estimated entropy as we decreased the amount of samples, removing in each iteration the sample which is the furthest from the samples’ center of mass and taking the mean after 100 runs. The experiment was repeated for different data dimensionalities. Although the results of the experiment are not shown here due to the lack of space, we summarize them here.

For Gaussian data, the k-d partition algorithm approximates better the trend of histogram based entropy estimation, even in the case of higher data

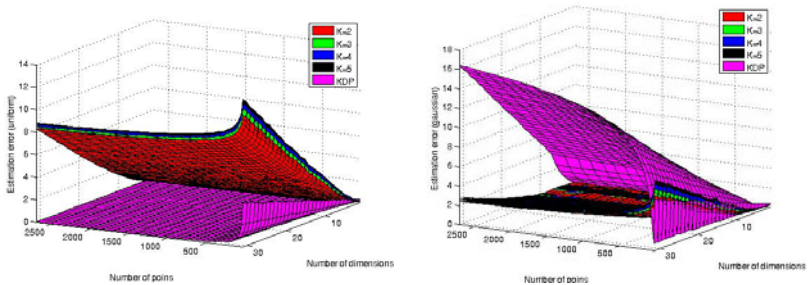


Fig. 1. K-d partition (KDP) and Leonenko *et al.* method (for $k = 2 \dots 5$) estimation bias for an uniform distribution in the range $[-3, 3]^d$ (left) and a Gaussian distribution with zero mean and $\Sigma = I$ (right)

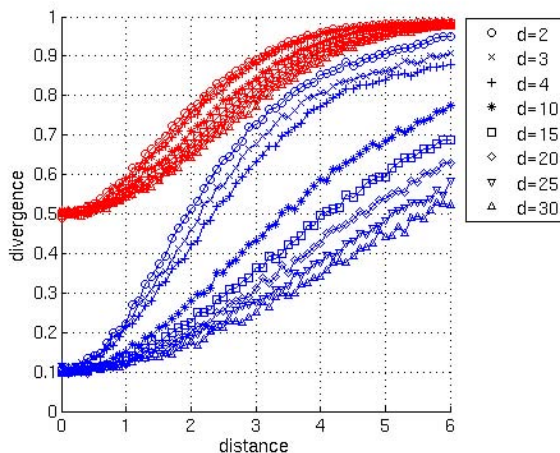


Fig. 2. Divergence estimation results using Friedman-Rafsky test (red) and k-d partition divergence (blue), for different data dimensionalities (d)

dimensionality. From $d = 3$, the Leonenko based estimation soon converges as N increases and, as a consequence, it has less discriminative power. For uniform data both estimators soon reach an asymptote; however, the k-d partition curve still approximates better to the shape of the histogram based curve.

4.2 Divergence Comparison

Now we compare the estimation results of our k-d partition based divergence with those of the Friedman-Rafsky test. Both methods were used to estimate the divergence of two sample sets gathered from two Gaussian distributions, starting with the same mean and variance, as we increased the distance between Gaussian centers until the probability that the samples overlap is low. The experiment was repeated for different data dimensionalities. The results are shown in Fig 2. In both cases, the divergence (y axis) increases with the distance between Gaussian centers (x axis). The values of Friedman-Rafsky test lie in the range $[0.5, 1]$. The range of values in the case of our k-d partition divergence is generally wider, but its results degrade for higher dimensionalities. However, even in the case of $d = 30$, the width of the range of values yielded by the k-d partition based divergence is similar to the one yielded by the Friedman-Rafsky test.

4.3 Number of Features

The amount of detected salient regions may have an effect on the quality and the repeatability of a feature extraction algorithm [2]. In the MDSS and the Kadir and Brady Scale Saliency algorithms we can set the percentage of most salient features to select, but not its final number due to the non-maximum suppression

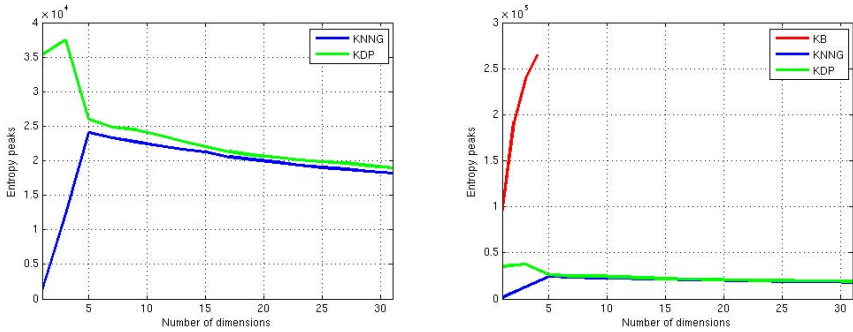


Fig. 3. Mean number of detected entropy peaks during Leonenko based MDSS (KNNG), k-d partition based MDSS (KDP) and Kadir and Brady Scale Saliency (KB) for increasing data dimensionality

step [1]. Thus, rather than performing a comparison of the two MDSS approaches based on the final number of detected features, our comparison was based on the number of entropic peaks found during the algorithm.

In Fig. 3 (left) we show the mean number of entropic peaks found during Leonenko and k-d partition based MDSS algorithms, using the images of the Bristol dataset, as we increased the number of dimensions (the number of layers used for entropy estimation). For a detailed description of the Bristol dataset see [7]. When data dimensionality is low, the results of the k-d partition based MDSS outperform those of the Leonenko based MDSS, providing a higher number of entropic peaks. For $d > 4$, although the number of detected entropic peaks is slightly higher in the case of the k-d partition based MDSS, the results are similar to those of the Leonenko based MDSS. Thus, both estimators may be considered equivalent in terms of number of detected features, and any of them could be applied to MDSS, if only this factor is relevant. In Fig. 3 (right) we compare the results of the MDSS and the Kadir and Brady Scale Saliency algorithms. It is unfeasible to apply the histogram based estimation for $d > 4$ due to the extremely high required computation time; thus, in Fig. 3 we are only showing partial comparison results. As can be seen, the Kadir and Brady Scale Saliency algorithm detects a higher amount of salient features in this range of data dimensions. This fact could be the cause of the better performance of this algorithm in the repeatability experiment in [7].

It must be noted that as the number of dimensions increase, the amount of entropy peaks decrease. This fact imposes a bound on the number of dimensions to which the MDSS can be applied. We tried, for instance, to apply the MDSS algorithm to 128D data images, in which a SIFT descriptor [12] was extracted for each image pixel, using a fixed scale. In most cases, the MDSS did not detect any entropy peak.

4.4 Choosing a MDSS Approach

Given the conclusions extracted during our experiments, and those in [7], the k-d partition entropy estimation algorithm should be preferred over the Leonenko estimator for MDSS implementation. Its computation time is remarkably lower, and it approximates better the trend of the Shannon's entropy for increasing saliency, even for high data dimensionality. Our new k-d partition based divergence also provides better estimation results than the KNNG based approach. The main drawback of both MDSS approaches is the low number of detected salient regions, that can decrease their performance in terms of repeatability. And although MDSS can cope with remarkably higher data dimensionality than the Kadir and Brady Scale Saliency algorithm, a bound on data dimensionality still exists, due to the fact that the number of features decrease as the number of dimensions increase.

5 A MDSS Application: Texture Categorization

In this section we show how the MDSS algorithm can be applied, in conjunction with the Lazebnik *et al.* [13] texture representation, to the texture categorization problem. In this problem, each image is showing one texture. We represent each texture image by a signature $S = \{(t_1, w_1), \dots, (t_n, w_n)\}$, where t_i is a texton and w_i is its relative weight. The steps required to build the signature following the Lazebnik method are: i) firstly, image features are extracted from grayscale intensities of the image, and a descriptor is computed for each feature, ii) agglomerative clustering is applied to all the descriptors of an individual image, and iii) the textons are the center of these clusters, and their relative weight is computed as the number of descriptors in the clusters divided by the total number of descriptors in the image. The obtained signatures can be compared by means of the Earth Mover's Distance (see [13] for more detail). In our case we apply MDSS to build a signature for each texture image from 15D data: all the pixels in the image are processed by means of a Gabor filter bank, consisting of 15 Gabor filters with different orientations and wavelengths. The Kadir and Brady Scale Saliency algorithm can not cope with this high dimensional data.

In Fig. 4 we show the results of our texture retrieval experiment (along with the output of the MDSS and the Scale Saliency algorithms for two example texture images). In this experiment, that shows the performance of a given texture representation, all images in the Brodatz dataset² are used as query image once. For each image query, we select images from the database in increasing order of EMD. The result is a plot that shows the average recall of all query images (being recall the number of images from the class of the query image retrieved so far divided by the total number of images in that class) versus the number of closest images retrieved. In Fig. 4 we compared the performance of the grayscale Scale Saliency and k-d partition based MDSS for the case of different descriptors: using only RIFT (*kadirrift* and *kdpeerift*, respectively), only spin

² <http://www.ux.uis.no/~tranden/brodatz.html>

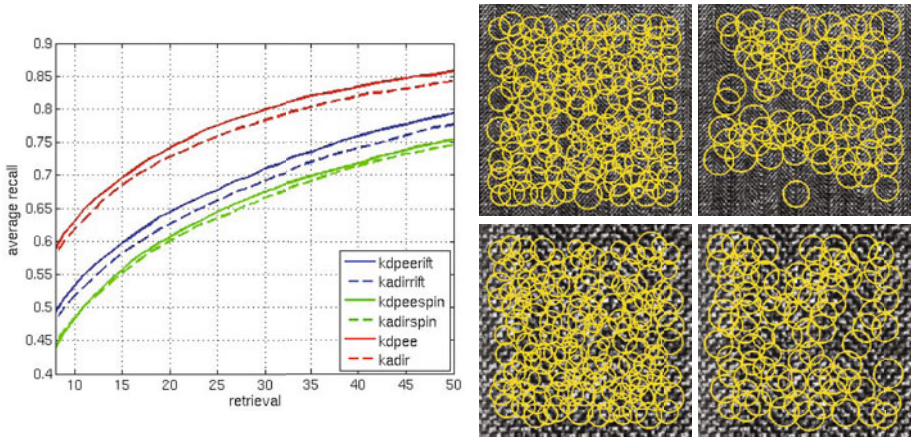


Fig. 4. Left: results of the texture categorization experiment. Right: output of the MDSS algorithm from 15D data (left) and the Scale Saliency algorithm from grayscale intensities (right), for two example texture images. In both cases the 150 most salient features (after non maximum suppression) were selected.

images (*kadirspin* and *kdpeespin*), and combining RIFT and spin images (*kadir* and *kdpee*). For a complete description of the RIFT and spin image descriptors, see [13]. In order to combine RIFT and spin images in the retrieval task, the total distance between two images is computed adding the normalized EMDs estimated for each individual descriptor.

Multi-dimensional data increased the performance of the texture retrieval task for each tested descriptor. However, its impact is not as noticeable as the impact of choosing an adequate descriptor. As can be seen in Fig. 4, the average retrieval is strongly affected by this last factor. The worst results are obtained for the case of spin images. RIFT increases the average recall, but the most significative improvement is achieved when combining both.

6 Conclusions and Future Work

The Scale Saliency algorithm by Kadir and Brady can be easily extended to process multi-dimensional data. However, its computational efficiency remarkably decreases with data dimensionality. We assess two approaches of MDSS based on different entropy and divergence metrics which computational order is linear with respect to data dimensionality. Our analysis shows that the k-d partition approach should be preferred over the graph based approach. We introduced a new divergence estimation method based on the k-d partition algorithm and the total variation distance, and we experimentally demonstrated its suitability. Finally, we showed a practical application of our approach in the context of texture categorization.

Our future work is addressed to evaluate the application of multi-dimensional data in other computer vision problems, like video processing or image retrieval. In the texture categorization context, we should also study the impact of using different Gabor filter banks, or even different input data. This is a combinatorial problem that may be treated with Machine Learning methods like feature selection.

References

1. Kadir, T., Brady, M.: Scale, Saliency and Image Description. *Int. J. Comp. Vision* 2, 83–105 (2001)
2. Mikolajczyk, K., Tuyelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. *Int. J. Comp. Vision* 65(1/2), 43–72 (2005)
3. Mikolajczyk, M., Leibe, B., Schiele, B.: Local features for object class recognition. In: *Proc 10th IEEE Int. Conf. Comp. Vision*, vol. 2, pp. 1792–1799 (2005)
4. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. *Int. J. Comp. Vision* 79(3), 299–318 (2008)
5. Newman, P., Ho, K.: SLAM-loop closing with visually salient features. In: *Proc. IEEE Int. Conf. Rob. Aut.*, pp. 635–642 (2005)
6. Suau, P., Escolano, F.: Multi-dimensional scale saliency feature extraction based on entropic graphs. In: *Proc. of the 4th Int. Symp. on Visual Computing*, vol. 2, pp. 170–180 (2008)
7. Suau, P., Escolano, F.: A new feasible approach to multi-dimensional scale saliency. In: *Proc. 11th Int. Conf. on Advanced Concepts for Intel. Vision Systems*, pp. 77–88 (2009)
8. Kozachenko, L., Leonenko, N.: On statistical estimation of entropy of a random vector. *Problems of Inf. Transm.* 23, 95–101 (1987)
9. Stowell, D., Plumbley, M.D.: Fast multidimensional entropy estimation by k-d partitioning. *IEEE Signal Processing Letters* 16(6), 537–540 (2009)
10. Denuit, M., van Belleghem, S.: On the stop-loss and total variation distances between random sums. *Stat. and Prob. Letters* 53, 153–165 (2001)
11. Cover, T., Thomas, J.: *Elements of Information Theory*. John Wiley & Sons, Chichester (1991)
12. Lowe, D.: Object recognition from local scale-invariant features. In: *Proc. of the 7th IEEE Int. Conf. on Comp. Vision*, vol. 2, pp. 1150–1157 (1999)
13. Lazebnik, S., Schmid, C., Ponce, J.: A sparse texture representation using local affine regions. *IEEE Trans. Pat. Anal. Mach. Intel.* 27, 1265–1278 (2005)