Pattern Recognition Letters 69 (2016) 72–81

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Robust histogram-based image retrieval☆

Cyril Höschl IV, Jan Flusser*

Institute of Information Theory and Automation, Czech Academy of Sciences, Pod vodárenskou věží 4, 182 08 Praha 8, Czech Republic

ARTICLE INFO

Article history:
Received 27 March 2015
Available online 12 November 2015

Keywords:
Image retrieval
Noisy image
Histogram
Convolution
Moments
Invariants

ABSTRACT

We present a histogram-based image retrieval method which is designed specifically for noisy query images. The images are retrieved according to histogram similarity. To reach high robustness to noise, the histograms are described by newly proposed features which are insensitive to a Gaussian additive noise in the original images. The advantage of the new method is proved theoretically and demonstrated experimentally on real data.

1. Introduction

Since the appearance of the first image databases in the 80's, image retrieval has been the goal of intensive research. Early methods did not search the images themselves but utilized some kind of metadata and image annotation (tagging) to retrieve the desired images. As many large-scale databases do not contain any annotations (manual annotation is expensive and laborious while automatic tagging is still under development), content-based image retrieval (CBIR) methods have become one of the most important challenges in computer vision. By CBIR we understand methods that search a database and look for images which are the "most similar" (in a predefined metric) to a given query image. CBIR methods do not rely on a text annotation and/or other metadata but analyze the actual content of the images. Each image is described by a set of features (often hierarchical or highly compressive ones), which may reflect the image content characteristics the user prefers – colors, textures, dominant object shapes, etc. The between-image similarity is then measured by a proper (pseudo) metric in the corresponding feature space.

CBIR is a subjective task because there is no "objective" similarity measure between the images. Hence, many CBIR systems aim to retrieve images which are perceived as the most similar to the query image for a majority of users and the users feel this similarity at the first sight without a detailed exploration of the image content. This requirement, along with the need for a fast system response, has led to a frequent utilization of low-level lossy features based on image colors/graylevels. A typical example is an intensity or color histogram. It is well known that the histogram similarity is a salient property for human vision. Two images with similar histograms are mostly perceived as similar even if their actual content may be very different from each other. On the other hand, those images that have substantially different histograms are rarely rated by observers as similar. Another attractive property of the histogram is that, if normalized to the image size, it does not depend on image translation, rotation and scaling, and depends only slightly on elastic deformations. Thanks to this, one need not care about image geometry and look for geometric invariants. Simple preprocessing can also make the histogram insensitive to linear variations of the contrast and brightness of the image. Hence, the histogram established itself as a meaningful image characteristic for CBIR [7–9].

The histogram is rarely used for CBIR directly as it is basically for two reasons. The histogram is not only an inefficiently large structure (in case of color images, the RGB histogram is stored in a vector of 2^{24} integers, which may be even more than the memory requirement of the original image) but it is also redundantly detailed. It is sufficient and computationally efficient to capture only the prominent features of the histogram and suppress the insignificant details. To do so, some authors compressed the histogram from the full color range into few bins [3,4] while some others represented the histogram by its coefficients in a proper functional basis. The advantage of the latter approach is that the number of coefficients is a user-defined parameter – we may control the trade-off between a high compression on one hand and an accurate representation on the other hand. It is very natural to get inspired by a clear analogy between histogram of an image and a probability density function (pdf) of a random variable. In probability theory, the pdf is usually

☆ This paper has been recommended for acceptance by Nappi Michele.

* Corresponding author. Tel.: +42 2 6605 2357; Fax: +42 2 6641 4903.
E-mail addresses: hoschl@utia.cas.cz (C. Höschl IV), flusser@utia.cas.cz (J. Flusser).

http://dx.doi.org/10.1016/j.patrec.2015.10.012
0167-8655/

characterized by its moments, so it is worth applying the same approach in the histogram-based CBIR [6,10].

The CBIR methods based on comparing histograms are sensitive to noise in the images, regardless of the particular histogram representation. Additive noise results in a histogram smoothing, the degree of which is proportional to the amount of noise. This immediately leads to a drop of the retrieval performance because different histograms tend to be more and more similar to each other due to their smoothing. In digital photography, the noise is unavoidable. When taking a picture in low light, we use high ISO and/or long exposure. Both amplifies the background noise, which is present in any electronic system, such that the noise energy may be even higher than that of the signal. Particularly compact cameras and cell-phone cameras with small-size chips (i.e. devices which produce vast majority of photographs on Flickr, on other servers, and on personal websites) suffer from this kind of noise, along with an omnipresent thermal noise. In-built noise reduction algorithms are able to suppress the noise only slightly and perform at the expense of fine image details.

Although the noise in digital photographs is an issue we can neither avoid nor ignore, very little attention has been paid to developing noise-resistant CBIR methods. The authors of the papers on CBIR have either skipped this problem altogether or rely on denoising algorithms applied to all images before they enter the database. Such a solution, however, is not convenient or even not realistic, because the denoising inevitably introduces artifacts such as high-frequency cutoff, requires additional time, and mostly also needs a cooperation of the user in tuning the parameters. In this paper, we present an original histogram-based image retrieval method which is not only robust but totally resistant (at least theoretically) to additive Gaussian noise. The core idea of the method is a proper representation of the histogram by certain characteristics, which are not affected by the noise. We stress that the paper does *not* aim to evaluate in which tasks and for what purposes a histogram-based CBIR is appropriate. We rather show how, if it is appropriate, it should be implemented in the case of noisy database and/or noisy query images. Our method does not perform any denoising and cannot replace it in the applications where the noise should be suppressed to improve the visual quality of the image.

In the rest of the paper, we first describe the noise model we are working with and show how this noise influences the image histogram. Then we present a noise-resistant representation of the histogram and demonstrate the advantage of this representation in CBIR. In the experimental part, we compare the new method with several traditional approaches and demonstrate their advantages on a database of more than 70,000 images and 30,000 queries.

2. The noise model

As we already mentioned, we primarily consider the thermal noise and electronic background noise of consumer cameras. It is a common belief that such noise n can be modeled as a stationary additive Gaussian white noise (AGWN) with zero mean and standard deviation σ, and that the noise is not correlated with the original image f. If this assumption were true, the noise normalized histogram h_n would have a Gaussian form

$$h_n(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{t^2}{2\sigma^2}\right), \tag{1}$$

where t is the index of the graylevel. The histogram h_g of the noisy image $g = f + n$ would then be a convolution of the original histogram and the noise histogram

$$h_g(t) = (h_f * h_n)(t).$$

Apparently, such an ideal model can hardly be encountered in practice. Let us however demonstrate on an example that it performs a reasonable approximation of a real noise. In Fig. 4(a), we can see a clip of size 427 × 386 pixels of a real noisy image taken under low-light conditions. In order to separate f and n, we took this image repeatedly twenty-times and we estimated f by time-averaging these 20 frames (see Fig. 4(b)). This allows us to calculate all three histograms h_g, h_f, and h_n and a synthetic histogram $h_c = h_f * h_n$ (see Fig. 1 from top to bottom). We can see that the noisy picture histogram in Fig. 1(c) matches the synthetic histogram in Fig. 1(d). Additionally, in Fig. 2 we can see the normality plot of the image noise n is very close to a normal distribution. We repeated this experiment for many images with the same conclusion. Hence, we consider our noise model acceptable and use it for deriving a proper histogram representation.

3. Histogram representation resistant to image noise

In this section, we present a representation of the image histogram by descriptors which are not affected by AGWN. These descriptors are based on the statistical moments of the histogram, which is a common approach to the characterization of pdf's in probability theory. Let h be a pdf of a random variable X. Then the quantity

$$m_p^{(h)} = \int x^p h(x) dx \tag{2}$$

where $p = 0, 1, 2, \ldots,$ is called *general moment* of the pdf. Clearly, $m_0 = 1$, m_1 equals the mean value and m_2 would equal the variance (if the histogram was centralized) of X. In general, the existence (finiteness) of the moments is not guaranteed, however if h is a (normalized) histogram, its support is bounded and all m_p's exist and are finite. On the other hand, any compactly-supported pdf can be exactly reconstructed from the set of all its moments.[1] In this sense moments provide a complete and non-redundant description of a pdf/histogram.

Unfortunately, the histogram moments themselves are affected by image noise. As the histogram of the noisy image is a smoothed version of the original histogram, it holds for its moments

$$m_p^{(g)} = \sum_{k=0}^{p} \binom{p}{k} m_k^{(n)} m_{p-k}^{(f)}. \tag{3}$$

This assertion can easily be proved just using the definitions of moments and of convolution. Since the noise is supposed to be Gaussian, h_n has a form of (1) and its moments are

$$m_p^{(n)} = \sigma^p (p-1)!! \tag{4}$$

for any even p. The symbol $k!!$ means a double factorial, $k!! = 1 \cdot 3 \cdot 5 \ldots k$ for odd k, and by definition $(-1)!! = 0!! = 1$. For any odd p the moment $m_p^{(n)} = 0$ due to the symmetry of the Gaussian distribution. Hence, (3) obtains the form

$$m_p^{(g)} = \sum_{k=0}^{[p/2]} \binom{p}{2k} \sigma^{2k} (2k-1)!! \cdot m_{p-2k}^{(f)}. \tag{5}$$

We can see that the moment of the noisy image histogram equals the moment of the clear image histogram plus some additional terms consisting of the moments of h_f of lower orders multiplied by a certain power of σ. For the first few moments we have

$$m_1^{(g)} = m_1^{(f)},$$

$$m_2^{(g)} = m_2^{(f)} + \sigma^2,$$

$$m_3^{(g)} = m_3^{(f)} + 3\sigma^2 m_1^{(f)},$$

[1] A more general moment problem is well known from theory of probability: can a given sequence be a set of moments of some compactly-supported function? The answer is yes if the sequence is completely monotonic.

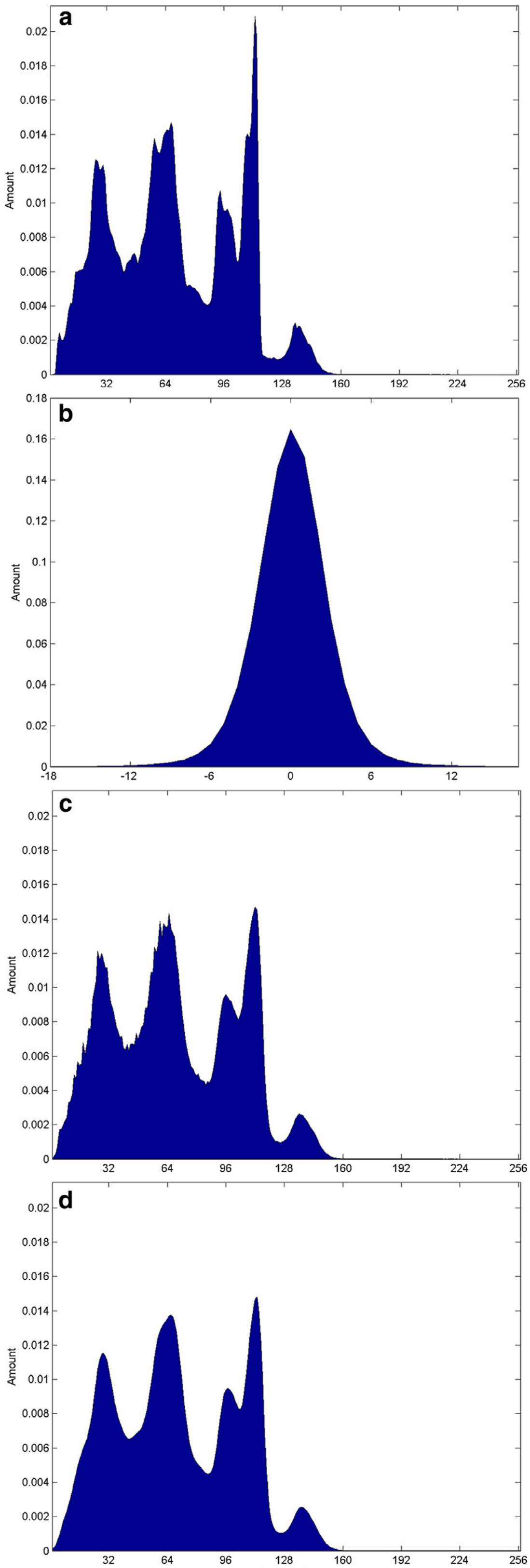

Fig. 1. Histograms of individual components of the captured image. Histogram h_f of the clear image (a), histogram h_n of the extracted image noise (note clear Gaussian shape) (b), histogram h_g of an originally captured noisy image (c), and the synthetic histogram h_c created as a convolution of the clear image histogram with the noise histogram (d). Notice the similarity of the noisy image histogram h_g and the synthetic histogram h_c.

$$m_4^{(g)} = m_4^{(f)} + 6\sigma^2 m_2^{(f)} + 3\sigma^4,$$

$$m_5^{(g)} = m_5^{(f)} + 10\sigma^2 m_3^{(f)} + 15\sigma^4 m_1^{(f)}.$$

To obtain noise-resistant descriptors, we have to eliminate the parameter σ. This can be done in a recursive manner, which leads to the definition of our histogram features

$$I_p = m_p - \sum_{k=1}^{[p/2]} (2k-1)!! \cdot \binom{p}{2k} I_{p-2k} m_2^k. \tag{6}$$

I_p can be equivalently expressed in a non-recursive form

$$I_p = \sum_{k=0}^{[p/2]} (2k-1)!! \cdot \binom{p}{2k} m_{p-2k} (-m_2)^k. \tag{7}$$

For any integer $p \geq 0$, the descriptor I_p is fully independent of the image noise regardless of the noise variance. In other words, the I_p value of an arbitrary noisy instance is the same as that of the original, and can be calculated without any denoising or estimating the noise variance (for the proof of this assertion see Appendix).

We use I_p values as histogram features for CBIR. Along with their resistance to noise, they provide an “almost complete” representation of the histogram. Having a full sequence of $I_p, \quad p = 1, 2, \ldots,$ we can recover from (7) all moments of the original histogram except $m_2^{(f)}$. This has a profound reason – since I_p is insensitive to noise, we cannot in principle recover the noise parameter σ, which influences $m_2^{(g)}$. Hence, we could recover the shape of the image histogram while its variance is a free parameter. This also corresponds to the fact that for any image $I_2 = 0$ while all other I_p's are valid. In other words, the full sequence of I_p's provides as much information about the image as its histogram itself with one degree of freedom allowing to incorporate an arbitrary unknown Gaussian smoothing of the histogram. In practice, we of course use only a finite set of these features, the number of which is determined by the user depending on the similarity of the images in the database – the more similar the images are to be discriminated, the more histogram features we need. For databases with dissimilar images, only a few (typically between 6 and 10) features are sufficient for histogram characterization, which provides an excellent compression ratio.

The intuitive meaning of the I_p's can also be understood as follows. The joint null-space of all I_p's is formed by all Gaussians, so the I_p's define the “distance” between the given histogram and the nearest Gaussian distribution. Equivalently, the I_p's actually measure the non-Gaussian component of the histogram.

It should be pointed out that the existence of such features that stay constant under a convolution of the histogram with a family of parametric kernels is a very rare phenomenon. The necessary (but not sufficient) condition is that this family must be closed with respect to convolution. In probability theory, such distributions are called *stable distributions*[2] and only three stable distributions are known in terms of elementary functions – Gaussian, Cauchy and Levy distributions.[3] Among them, only the Gaussian distribution has all finite moments, so our moment-based approach can hardly be extended to any other noise model.

It is worth mentioning that all above equations remain valid if we use *central moments* of the histogram instead of the general ones. In that way we achieve an invariance of the method to the overall brightness of the images without any histogram normalization.

[2] Equivalently, this property can be formulated such that the sum of two independent random variables, whose distributions belong to the family, has a distribution also from this parametric family.

[3] Even the generalized Gaussian distribution is not stable for exponents other than 2.

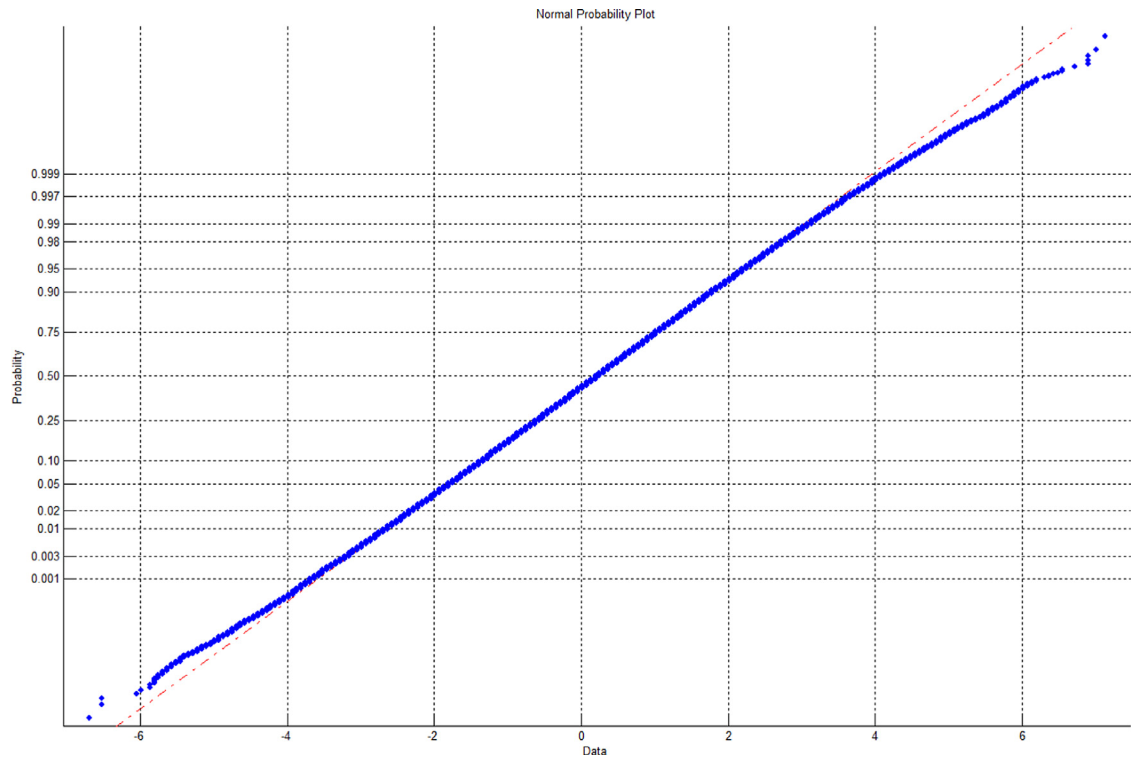

Fig. 2. Normality plot of a camera noise. Blue points indicate the noise data, the red line shows the estimated normal distribution. If the data was precisely from the normal distribution then the blue marks would be linear. Overlap of the blue marks with the dashed red line shows that the data is almost normal. The deviations on both ends are due to the value cut-off in the image. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article).

4. Experiments

4.1. Invariance on simulated AWGN

In the first experiment we demonstrate the invariance property of I_p (7) on pictures with simulated noise. We use a testing database of 1,000 pictures randomly gathered from Flickr[4]. The average picture size of is 1.3 Mpx and all pictures were converted to grayscale levels.

For each sample picture in the database, we created its noisy version by adding a zero-mean Gaussian white noise of various variances. It should be noted that even though the original grayscale image values range from 0 to 255, we do not cut off the values of noisy image so they can range from negative values to values higher than 255.

For each picture and each signal-to-noise ratio (SNR), we extracted two histograms: h_f of the original image and h_g of its noisy version. To show that the invariants I_p give the same results for both clear and noisy pictures, we calculated the ratio

$$r = \frac{I_p^{(f)}}{I_p^{(g)}}, \tag{8}$$

where we have applied the invariant function (7) on the histogram of the original image f divided by the invariant applied on the histogram of the noisy image g. In Fig. 3 we show the distribution of ratio r for invariants of orders $p = 3, 6, 10$ and 10 different SNRs from 5 to 32. It can clearly be observed that a majority of the ratios is almost equal to 1. It is also evident that the variance of the distribution of r increases as the SNR decreases. The fact that the ratio is not precisely 1 for all cases is because the randomly generated noise is not always exactly Gaussian. Distributions for all three chosen invariant orders are quite similar. However, the higher the order of the invariant function, the more significant the influence of the numerical errors. This can be observed as a higher variance of the distributions in the higher-order boxplots. This is an experimental verification that I_p is invariant under ideal Gaussian noise.

4.2. Invariance on real pictures

In the second experiment we demonstrate the invariance of (7) on photographs captured by a compact camera[5]. This is a much more challenging situation namely because of the value cut-offs, which violate the normality of the noise distribution.

We captured 20 different scenes under various light conditions. The light was always low to get a noticeable noise and by light changes we controlled (at least roughly) the noise variance. The estimated SNR was between 15 and 20. We took each scene 20 times and then we estimated the clear image by time-averaging, since under low light it was impossible to obtain a clear image directly (see Fig. 4 for an example).

As in the previous experiment, we evaluated the ratio (8) of invariant functions on histograms of noisy and clear pictures. To show the invariance property, the ratio r should be close to 1. Unlike the simulated noise, the real camera noise is subject to cut-off and the histogram support is bounded by the values 0–255. This causes the input data for (7) not to meet the required theoretical assumptions perfectly. In any case, the results of the invariants are quite satisfactory as we can see in Fig. 5. The median of the ratios is almost equal to 1 for all chosen invariant orders $p = 3, \ldots, 10$ and furthermore, a majority of invariant ratios is very close to 1. For a comparison and to show that this property is far from being obvious, we also calculated the same ratios for the histogram moments themselves. As one can

[4] In all our experiments we use original photographs without any postprocess modifications. Pictures are from the set used by the authors of [5].

[5] SONY Cyber-Shot DSC-H50, the resolution 3.1 Mpx was used.

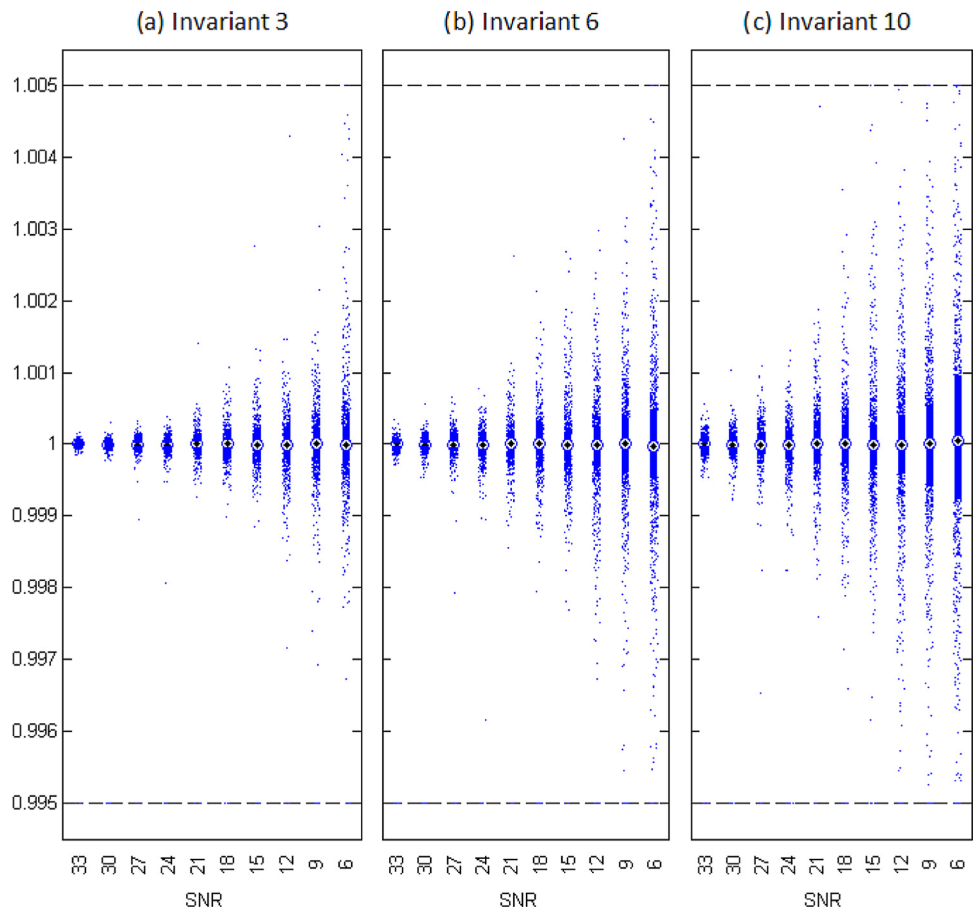

Fig. 3. The boxplots show the distribution of 1,000 ratios of invariants calculated on original images and their noisy versions. The boxplots from left to right show the results for invariant orders 3, 6 and 10 respectively. The central mark shows the median, thick bar depicts 50% of the data between 25th and 75th percentiles. Outliers outside this range are marked as dots.

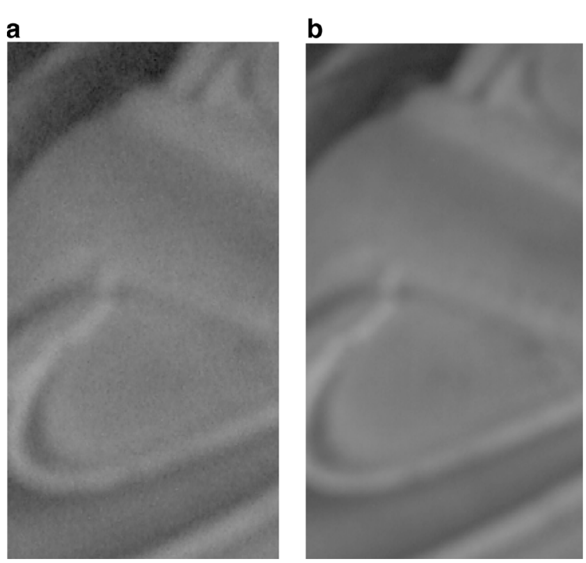

Fig. 4. A crop of the scene photographed in low light. Originally captured noisy image (a) and the noise-free image constructed by averaging 20 noisy frames of the same scene (b).

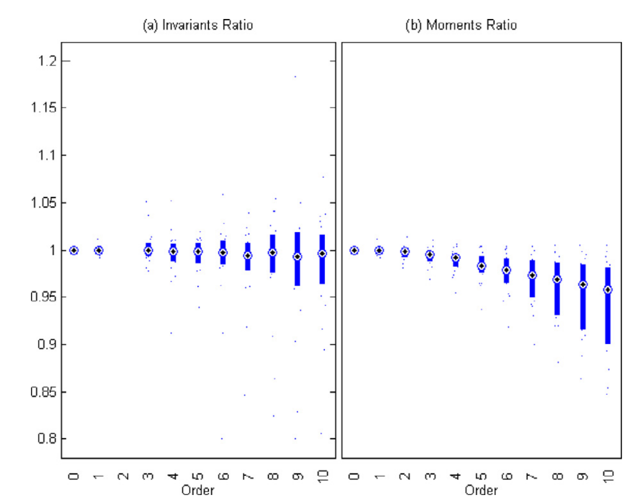

Fig. 5. (a) The boxplots show the ratio (8) of invariants calculated on histograms of real clear and noisy images. Central mark is the median of the distribution. Thick bar depicts 50% of the data between 25th and 75th percentiles. Outliers outside this range are marked as dots. (b) The boxplots show the same ratio where plain moments m_p were used instead of invariants I_p. This graph illustrates that the histogram moments cannot be used instead of the invariants since they are heavily affected by the noise.

see in Fig. 5(b), their behavior is dramatically different. The plain histogram moments are affected heavily by the noise and their relative error increases as the order grows. They do not exhibit any invariance to noise. Hence, this experiment shows that the invariants actually bring a significant added value.

4.3. Image retrieval

Content-based image retrieval is a challenging task where the user selects a query image to retrieve a list of “similar” images (the similarity measure is pre-defined by the user, here we measure the similarity by image histograms) from a large database of pictures. The natural requirement is to avoid mismatches where the CBIR method returns images that are not related to the query image. For the human perception, two images with the same content seems similar even though one of them is affected by noise. On the other hand, CBIR methods based on comparing image histograms are sensitive to noise that modifies the histogram (see Fig. 7) and therefore standard methods may produce many mismatches. If the database system contains pictures of a similar histogram and either the input query image or the database images are affected by noise, then the danger of mismatches is high.

The aim of this experiment is to show a practical application of the proposed invariants (7) to CBIR. In this experiment the database contained clear images (or at least images with invisible noise) while the query image was always a noisy version of one database image. To make the task challenging, we intentionally included pictures of similar histograms into the database. We randomly gathered 71,842 photographs from Flickr and categorized them into 314 clusters based on histogram similarity. The image clustering was used here only to select the images for the experiments. To save time, we always limited the search to the respective cluster only. Each query image was matched only against that cluster of images with similar histograms. Matching to dissimilar histograms does not make sense because all methods correctly reject such trials. It should be noted that the similarity of the histograms may or may not correspond to the visual similarity between the images. In Fig. 6(a) we can see an example of visually different images while Fig. 6(b) provides an example of very similar images. In both cases, the histograms inside the groups are similar.

We performed 31,400 queries to achieve a statistical significance. We created query images of five SNR levels (5, 10, 15, 20 and 25). For each SNR we generated 20 different instances of the noise. The histograms of the query images were heavily smoothed due to the noise (see Fig. 7). Each query was independently answered by the following methods.

- The above mentioned invariants $I_1, \ldots, I_{10}$ calculated from the histogram of the *graylevel* image. To convert the original color images into graylevels, the Matlab function `rgb2gray` was used. The database image with the minimum Euclidean distance was retrieved. Since the invariants grow rapidly as the order p increases, we normalized the I_p's to keep them in a comparable range before calculating the distance. This method is referred to as *Invariants Gray*.
- Invariants applied on the color image channel-wise and subsequently concatenated. The feature vector was $I_1^R, \ldots, I_{10}^R, I_1^G, \ldots, I_{10}^G, I_1^B, \ldots, I_{10}^B$. The rest of the method was the same as in the previous case. This method is referred to as *RGB Vectors*.
- Invariants applied on the color image channel-wise. The concatenation was replaced by a voting scheme. The distance is calculated for each channel separately and a majority vote is applied. If at least two channels vote for the same database image, this image is retrieved. No image is retrieved if each channel votes for different database image. This method is referred to as *RGB Vote*.
- A method similar to the first one but instead of using invariants, we used plain moments $m_0, m_1, \ldots, m_{10}$ of the (graylevel) histogram. This method is referred to as *Moments Gray*.
- Full histogram matching. In this method, we match a complete graylevel histogram (256 bins) by the minimum Euclidean distance. This method is referred to as *Histogram*.
- The last method is the only one that contains denoising as a preprocessing. We denoised the query images first by a wavelet-based denoising [1] and then applied full histogram matching as in the *Histogram* method. This method is referred to as *Denoised*.

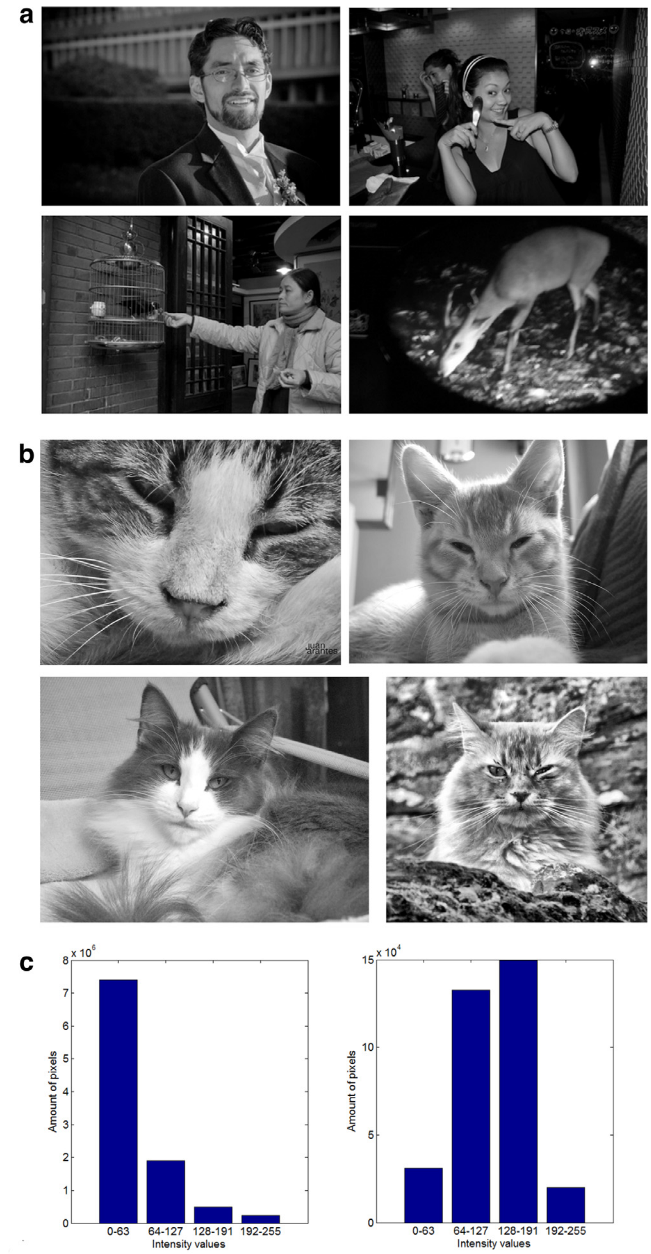

Fig. 6. Sample images from the test database. Pictures were clustered according to their histogram similarity. When considering histograms simplified into four bins, all pictures within one cluster have the same simplified histogram. In (a) and (b) there are previews of pictures from two clusters with corresponding histograms in (c) (on the left is a histogram for cluster (a) and on the right for cluster (b)). Some clusters contain pictures that have the same histogram but look differently (e.g. cluster (a)), some clusters contain pictures that look similar (e.g. cluster (b)).

Since we know the ground truth, we can evaluate the correct retrieval rate. Fig. 8 shows the results of retrieval for all the methods as a function of the SNR. The results mostly confirmed our theoretical expectation.

The *RGB Vectors* performed best, followed by the *Invariants Gray* method. The overall performance of both is very good. The *RGB Vote* performs slightly worse, which may look a bit surprising. The reason is that the majority vote from three votes is very strong criterion (we actually decide on the 2/3 majority and not on the absolute majority) and that is why we miss some correct matches. Since the *Invariants Gray* is three times faster, it may be an optimal compromise for large-scale tasks. The difference between these three

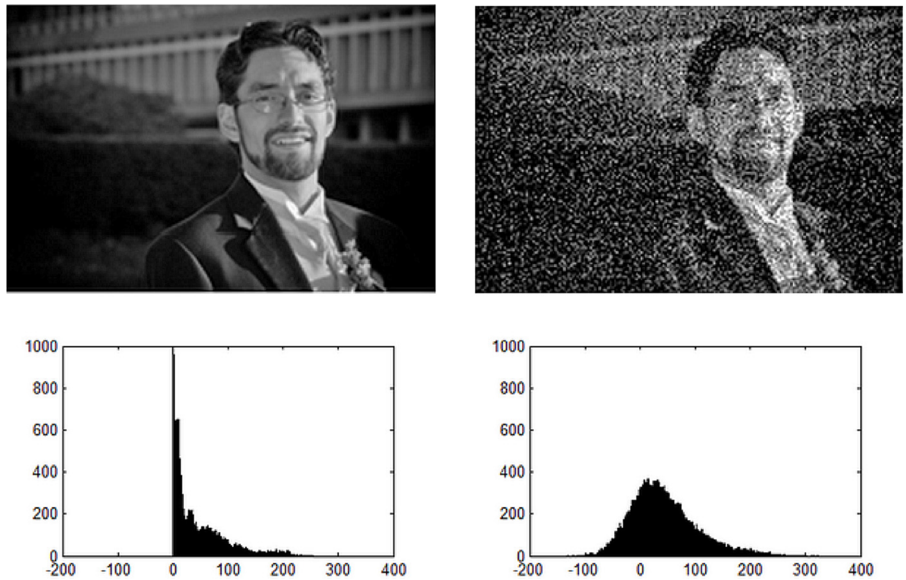

Fig. 7. Example of the query image affected by noise (SNR = 5) (right) and the clear version of the image in the database (left). On the bottom there are histograms of the images. It can be seen that the noise causes significant modification of the histogram.

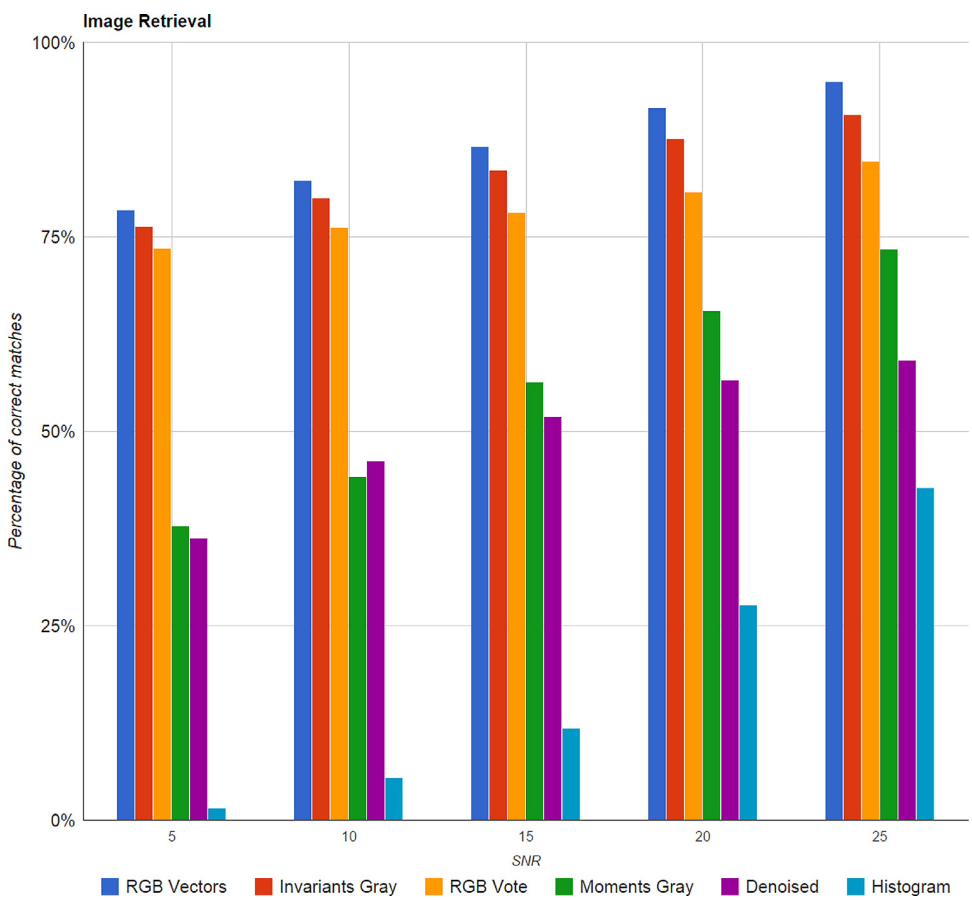

Fig. 8. Image retrieval experiment results. The graph shows percentages of correct matches for 6,280 queries for each SNR (total is 31,400 queries in 314 databases). The methods based on moment invariants outperform the methods based on plain moments or histogram matching.

methods and the other methods increases as the SNR decreases. The plain moments perform better than a complete histogram matching. The explanation is that we used only 10 low-order moments that describe global characteristics of the histogram which are less influenced by the noise than the complete histogram itself.

The most surprising result is the poor performance of the *Denoised* method. The reason is that the denoising decreases the noise level in the image but does not restore the original histogram well. It should be noted that we did not use the knowledge of the SNR when setting the parameters of the denoising algorithm. Another serious drawback of this approach is that it requires a significant extra time to perform the denoising. We also tried to replace the wavelet denoising by BM3D algorithm [2], which is one of the highest rated existing denoising methods and re-run the experiment. However, the BM3D

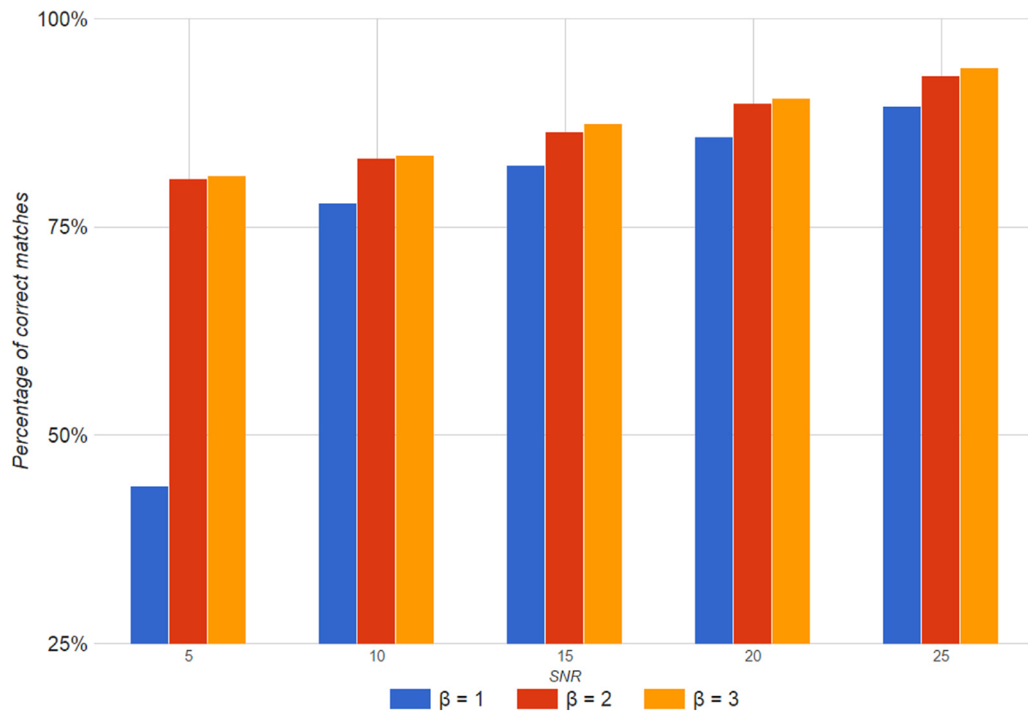

Fig. 9. Retrieval of images corrupted by heavy-tailed ($\beta = 1$), Gaussian ($\beta = 2$), and light-tailed ($\beta = 3$) noise. The graph shows percentages of correct matches by means of histogram invariants for 71,842 queries for each SNR.

Table 1
Picture database summary.

Number of databases	314
Total number of pictures	71,842
Number of queries	31,400
Average pictures count per DB	229

algorithm is so slow (10 min for one query image with 20 instances of noise) that we ran it on a small subset only to conclude that the success rate is comparable to that of the wavelet denoising. Hence, an interesting conclusion is that denoising followed by histogram matching is absolutely not suitable in terms of both success rate and speed, regardless of the particular denoising algorithm.

Finally, we tested the performance of the proposed method in the presence of noise which is not exactly Gaussian. We generated the noise underlying the *generalized normal distribution*

$$h_{\alpha,\beta}(t) = \frac{\beta}{2\alpha\Gamma(1/\beta)} \exp\left(-\frac{|t|}{\alpha}\right)^{\beta}, \tag{9}$$

This probability density function is equivalent to the Gaussian distribution for $\beta = 2$. For $\beta < 2$, it has heavier tails and for $\beta > 2$ lighter tails than the Gaussian distribution. We used two distinct β-values in this test: $\beta = 1$, which yields the Laplace distribution, and $\beta = 3$. For a comparison, we included the Gaussian case $\beta = 2$. We run the experiment with these three β-values on 71,842 test images (Table 1). We successively generated a noise of SNR ranging from 5 to 25. The retrieval results achieved by the invariants of a grey-level histogram (the same method as *Invariants Gray* in the previous experiment) are summarized in Fig. 9. One can observe two noticeable trends. The noise distribution with lighter tails does not cause any problems. The retrieval rate is fully comparable (or even slightly better in some cases) to the Gaussian noise and the method could be applied to this type of noise, too. The heavy-tailed noise significantly decreases the performance of the method on low SNR levels. As the SNR increases, the performance approaches (logically) the performance achieved for Gaussian noise.

5. Extension to color histograms

The presented invariants can be extended from 1-D graylevel histograms to color or even multispectral histograms. A complete histogram of an image with N spectral/color bands is an array of 2^{bN} integers, where b is the number of bits used to encode the pixel intensity in one band (typically $b = 8$). The size of the multispectral histogram grows exponentially with N which makes it very inefficient for small images of many bands. As the histogram size does not depend on the image size, this representation can be useful for large images with a low band number, e.g. for traditional color images with $N = 3$. Assuming a Gaussian noise is added to each band, the N-D histogram of a noisy image is again a convolution of the original histogram and the N-D Gaussian density function, which is given as

$$h_n(\mathbf{t}) = \frac{1}{\sqrt{(2\pi)^N |C|}} \exp\left(-\frac{1}{2}\mathbf{t}C^{-1}\mathbf{t}'\right), \tag{10}$$

where $\mathbf{t} \equiv (t_1, \ldots, t_N)$ and C is the noise covariance matrix.

If C is diagonal, i.e. if the noise in any two spectral bands are mutually uncorrelated, then (10) is a product of 1-D Gaussians and we can easily derive N-dimensional analogies of the invariants (6) and (7)

$$I_{\mathbf{p}} = m_{\mathbf{p}} - \sum_{\substack{\mathbf{k}=\mathbf{0} \\ 0<|\mathbf{k}|}}^{[\mathbf{p}/2]} (2\mathbf{k}-\mathbf{1})!! \cdot \binom{\mathbf{p}}{2\mathbf{k}} I_{\mathbf{p}-2\mathbf{k}} \mathbf{m}_2^{\mathbf{k}} \tag{11}$$

$$I_{\mathbf{p}} = \sum_{\mathbf{k}=\mathbf{0}}^{[\mathbf{p}/2]} (2\mathbf{k}-\mathbf{1})!! \cdot \binom{\mathbf{p}}{2\mathbf{k}} (-1)^{|\mathbf{k}|} m_{\mathbf{p}-2\mathbf{k}} \mathbf{m}_2^{\mathbf{k}} \tag{12}$$

where the boldface characters are used for standard vector notation and $\mathbf{m}_2 \equiv (m_{20\ldots0}, m_{02\ldots0}, \ldots, m_{00\ldots2})$.

The assumption of C being diagonal seems to be natural and it actually holds for multispectral sensors where individual bands are captured independently, such as satellite and aerial scanners, and for

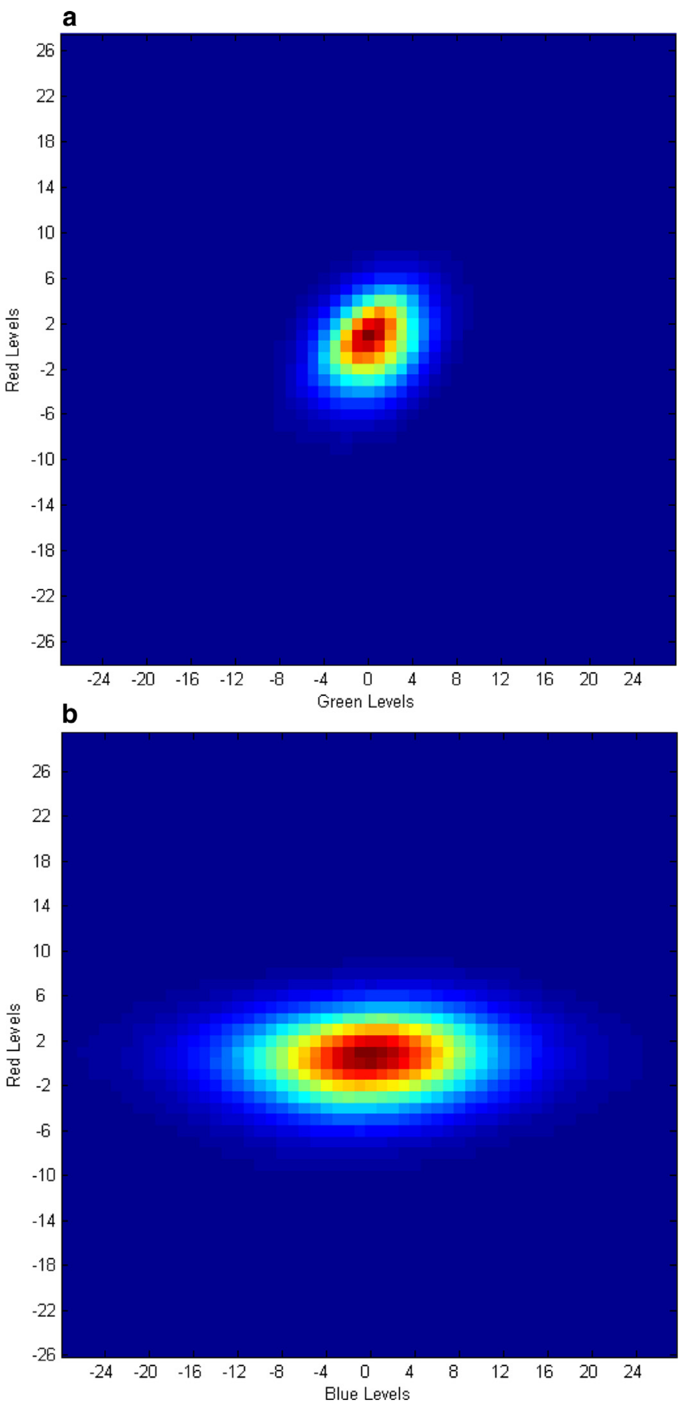

Fig. 10. An example of 2D histograms of the noise extracted from a real image. Red and green bands are correlated $\varrho = 0.33$, blue and red bands are almost independent $\varrho = 0.06$.

multimodal medical images. This is however not true for color images captured by single-chip consumer cameras, where the entering light is split into red, green and blue channels by a color filter array, most commonly arranged into Bayer pattern. To enhance the spatial resolution, the camera applies embedded interpolation algorithms on raw data (including the noise). This interpolation may introduce between-channel correlation not only of clear image data (where the correlation is expected anyway) but also of the noise components which theoretically should be independent. To illustrate this phenomenon, we extracted the noise components of a real color image using the same technique as described in Section 2 and visualized 2-D histograms of the noise (see Fig. 10). While the blue and the red bands are uncorrelated, the correlation between red and green is about $\varrho = 0.33$. We performed this test on several images with basically the same results (the particular values of course depend on the camera type, on the setting and on other conditions). Note also that the noise variances in individual channels typically differ from each other, but this is not a serious problem.

If C is not diagonal, it would still be possible to derive histogram features insensitive to such kinds of correlated noise, assuming that the eigenvectors of C are of a known orientation, which is constant for all images in question. We could rotate the histogram such that the noise becomes uncorrelated, which is always possible, and then proceed as described above. This is, however, not the case of real color noise, where the eigenvectors of C are basically random. Under such conditions, the invariant approach cannot be used correctly and we are limited to channel-wise histograms.

6. Conclusion

Histogram of a noisy image, both visual appearance and common numerical characteristics, are significantly affected by additive noise in the image. Provided the noise is Gaussian, we proposed original histogram descriptors which are invariant w.r.t. the noise. We proved that along with the theoretical invariance the descriptors are sufficiently robust on real images corrupted by thermal and electronic sensor noise. As demonstrated experimentally, the proposed descriptors can be used as the features in a histogram-based retrieval if the database and/or query images are heavily noisy and standard descriptors fail. We approved that the retrieval based on the new invariants significantly outperform the other more traditional methods included in our tests. We also proved that the method can be used even if the noise distribution is not exactly Gaussian, but has lighter tails.

Acknowledgment

This work has been supported by the Grant no. GA15-16928S of the Czech Science Foundation.

Appendix

In this Appendix we present a formal proof that the image features defined in Eq. (6) do not change under Gaussian white noise. To prove this, it is sufficient to show that they do not change if the image histogram is convolved by an arbitrary zero-mean Gaussian pdf (1) of an unknown parameter σ. We prove this by induction over p. The validity is trivial for $p = 0, 1, 2$ and can be verified easily for $p = 3$ by substitution of (5) into (6). Let us now prove (6) for an arbitrary $p > 3$ provided that it holds for all lower indices.

$$I_p^{(g)} = m_p^{(g)} - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} I_{p-2k}^{(g)} (m_2^{(g)})^k$$

where $K = [p/2]$. Using (5) and the assumption that $I_{p-2k}^{(g)} = I_{p-2k}^{(f)}$ we get

$$\begin{aligned}
I_p^{(g)} &= \sum_{k=0}^{K} \binom{p}{2k} \sigma^{2k} (2k-1)!! \cdot m_{p-2k} \\
&\quad - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} I_{p-2k} (m_2 + \sigma^2)^k \\
&= m_p - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} \\
&\quad \times \left(I_{p-2k} \sum_{j=0}^{k} \binom{k}{j} \sigma^{2j} m_2^{k-j} - \sigma^{2k} m_{p-2k} \right)
\end{aligned}$$

$$= I_p^{(f)} - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} \times \left(I_{p-2k} \sum_{j=1}^{k} \binom{k}{j} \sigma^{2j} m_2^{k-j} - \sigma^{2k} m_{p-2k} \right)$$

$$= I_p^{(f)} - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} I_{p-2k} \sum_{j=1}^{k} \binom{k}{j} \sigma^{2j} m_2^{k-j} + \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} \sigma^{2k} m_{p-2k}$$

$$= I_p^{(f)} - A_p + B_p.$$

In the above expressions, we dropped the index $^{(f)}$ for the sake of simplicity whenever it is clear from the context. Now we show that $A_p = B_p$, which will complete the proof. To express the double factorial, we use the relation $(2k-1)!! = (2k)!/2^k k!$.

$$A_p = \sum_{k=1}^{K} \sum_{j=1}^{k} (2k-1)!! \cdot \binom{p}{2k} \binom{k}{j} I_{p-2k} \sigma^{2j} m_2^{k-j}$$

$$= \sum_{j=1}^{K} \sum_{k=j}^{K} (2k-1)!! \cdot \binom{p}{2k} \binom{k}{j} I_{p-2k} \sigma^{2j} m_2^{k-j}$$

$$= \sum_{j=1}^{K} \sum_{k=0}^{K-j} (2(k+j)-1)!! \cdot \binom{p}{2(k+j)} \binom{k+j}{j} I_{p-2(k+j)} \sigma^{2j} m_2^{k}$$

$$= \sum_{j=1}^{K} \frac{p!\sigma^{2j}}{j!2^j} \sum_{k=0}^{K-j} \frac{m_2^k}{k!(p-2j-2k)!2^k} I_{p-2j-2k}$$

$$= \sum_{j=1}^{K} \frac{p!\sigma^{2j}}{j!2^j} \left(\sum_{k=1}^{K-j} \frac{m_2^k}{k!(p-2j-2k)!2^k} I_{p-2j-2k} + \frac{I_{p-2j}}{(p-2j)!} \right)$$

The inner sum equals, according to (6), to

$$\frac{m_{p-2j} - I_{p-2j}}{(p-2j)!}.$$

Hence,

$$A_p = \sum_{j=1}^{K} \frac{p!\sigma^{2j} m_{p-2j}}{j!(p-2j)!2^j} = B_p. \qquad \square$$

In a similar way, by means of induction over p, it is also possible to prove the equivalence between (6) and (7). We briefly show the induction step.

$$I_p = m_p - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} m_2^k I_{p-2k}$$

$$= m_p - \sum_{k=1}^{K} (2k-1)!! \cdot \binom{p}{2k} m_2^k \times \sum_{j=0}^{K-k} (2j-1)!! \cdot \binom{p-2k}{2j} (-m_2)^j m_{p-2k-2j}$$

$$= m_p - \sum_{k=1}^{K} \sum_{j=0}^{K-k} (-1)^j \frac{p!}{2^{k+j} k! j! (p-2k-2j)!} m_2^{k+j} m_{p-2k-2j}$$

$$= m_p - \sum_{k=1}^{K} \sum_{j=k}^{K} (-1)^{j-k} \frac{p!}{2^j k! (j-k)! (p-2j)!} m_2^j m_{p-2j}$$

$$= m_p - \sum_{j=1}^{K} \sum_{k=1}^{j} (-1)^{j-k} \frac{p!}{2^j k! (j-k)! (p-2j)!} m_2^j m_{p-2j}$$

$$= m_p - \sum_{j=1}^{K} (-1)^j \frac{p!}{2^j (p-2j)!} m_2^j m_{p-2j} \sum_{k=1}^{j} \frac{(-1)^k}{k!(j-k)!}$$

Since

$$\sum_{k=1}^{j} (-1)^k \cdot \binom{j}{k} = -1$$

for any j, we obtain

$$I_p = m_p + \sum_{j=1}^{K} (2j-1)!! \binom{p}{2j} (-m_2)^j m_{p-2j}$$

$$= \sum_{j=0}^{K} (2j-1)!! \binom{p}{2j} (-m_2)^j m_{p-2j},$$

which exactly matches Eq. (7).

References

[1] R. Coifman, D. Donoho, Translation-invariant de-noising, in: A. Antoniadis, G. Oppenheim (Eds.), Wavelets and Statistics, Lecture Notes in Statistics, vol. 103, Springer, New York, 1995, pp. 125–150, doi:10.1007/978-1-4612-2544-7_9.

[2] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, IEEE Trans. Image Process. 16 (8) (2007) 2080–2095, doi:10.1109/TIP.2007.901238.

[3] S. Jeong, Histogram-Based Color Image Retrieval., Technical Report, Stanford University, 2001. Psych221/EE362 Project Report.

[4] G.-H. Liu, L. Zhang, Y.-K. Hou, Z.-Y. Li, J.-Y. Yang, Image retrieval based on multi-texton histogram, Pattern Recognit. 43 (7) (2010) 2380–2389, doi:10.1016/j.patcog.2010.02.012.

[5] B. Mahdian, R. Nedbal, S. Saic, Blind verification of digital image originality: A statistical approach, IEEE Trans. Inf. Forensics Secur. 8 (9) (2013) 1531–1540, doi:10.1109/TIFS.2013.2276000.

[6] M. Mandal, T. Aboulnasr, S. Panchanathan, Image indexing using moments and wavelets, IEEE Trans. Consum. Electron. 42 (1996) 557–565.

[7] G. Pass, R. Zabih, Histogram refinement for content-based image retrieval, in: Proceedings of the Third IEEE Workshop on Applications of Computer Vision (WACV), 1996. '96., 1996, pp. 96–102, doi:10.1109/ACV.1996.572008.

[8] M.J. Swain, D.H. Ballard, Color indexing, Int. J. Comput. Vis. 7 (1) (1991) 11–32, doi:10.1007/BF00130487.

[9] W. Xiaoling, A novel circular ring histogram for content-based image retrieval, in: Proceedings of the First International Workshop on Education Technology and Computer Science (ETCS), 2009, vol. 2, 2009, pp. 785–788, doi:10.1109/ETCS.2009.437.

[10] P.-T. Yap, R. Paramesran, Content-based image retrieval using Legendre chromaticity distribution moments, IEE Proc. Vis. Image Signal Process. 153 (1) (2006) 17–24, doi:10.1049/ip-vis:20045064.