Perceptual Image Hashing: Tolerant to Brightness and Contrast Corrections Method Based on Cumulative Histogram Slicing

Aleksei Zhuvikin[1], Valery Korzhik[2]
The Bonch-Bruevich Saint-Petersburg
State University of Telecommunications
Saint-Petersburg, Russia
[1]mail@zhuvikin.com, [2]val-korzhik@yandex.ru

Abstract—**Perceptual image hashing is used in a wide range of practical applications which include content image authentication, digital watermarking, pattern recognition, computer vision and database fast duplicate image retrieval. Existing techniques are not well suited for the significant brightness and contrast corrections. The main point is that such manipulations can lead to information loss due to the histogram truncation in cases when pixel values are out of the dynamic range. In order to address the issue a novel technique is suggested. Cumulative histogram slices as a pivot for the subsequent image features calculations are used. The points of slicing are calculated in a way they are robust to content preserving manipulations such as brightness and contrast corrections. This approach allows one to handle situations when some of the content slices are lost due to the pixel value overflow. On the other hand, if one tampers image content within any existing slice it will then be detected by comparing the correspondent calculated and provided hash values. Experiment results show that the suggested method has sufficient sensitivity to detect image tampering whereas being tolerant to even significant brightness and contrast corrections. The memory consumption allows one to use the proposed method with the digital watermarking schemes.**

I. Introduction

There is a constant tendency for the increase of multimedia part in the global Internet traffic [1]. The reason is that such type of content is better perceived by a human mind due to its naturalness. Upcoming telecommunication technologies will achieve data bandwidth to transmit even more multimedia. However, there is a lack of protection and control over what the digital content is produced from and of how it is published and transmitted. The most popular type of the multimedia content is motionless images and videos. The latter can also be recognized as a sequence of motionless images. The more images are produced, the more demand for image processing and editing appears. Image editing software has become an extremely easy task. This leads to tampering and content modifications without the image copyright owner permission.

Moreover, there are some content-preserving manipulations which are even more widely used. In particular, almost every web service uses an image compression before transmitting it over the network or before saving it into the data volumes. These manipulations allow one to spend resources in an economical way. The other useful application such as digital watermarking embeds additional data into the image itself. The aim is to provide the information about image owner or to be the auxiliary data for content image authentication. Digital watermarking technique also introduces some noise in pixel values after embedding but it does not distorts an image content. Some content-preserving manipulations are purposed to improve image visual quality and include image resizing, noise reduction, filtering, brightness, contrast and gamma corrections. Being actually a part of technological cycle, these operations are not for the image content disruption but for the practical purposes.

Regardless the facts mentioned above, both image tampering and content-preserving manipulations introduce actual image changes. Some applications require to validate the images, find duplicates in the database and check the difference in order to distinguish between them. Conventional cryptographic hashing technique does not fit that purpose due to the avalanche effect. To address these issues so-called perceptual image hashing techniques are suggested in literature [2-12]. Perceptual image hashing is to extract the most significant content description features leaving them robust to content-preserving operations while being sensitive to image tampering. Such algorithms work with locality-sensitive hashing (LSH) techniques for the fast image duplicate retrieval by means of the database [13]. In order to be used in the image authentication and digital watermarking areas the algorithms have to provide an output sequence of notably short length. Finally, the general restriction for the perceptual image hashing algorithms is the relatively low calculation time.

One can find a brief overview of the already existing perceptual image hashing techniques and related works in Section II. The proposed method of cumulative histogram slicing and an example method of the feature extraction based on HWT are described in Section III. Evaluation of the proposed method effectiveness versus content-preserving operations and image tampering is given in Section IV. In the same section there is a comparison between state-of-the-art algorithms and the proposed one. We conclude the paper in Section V.

II. Related Works

There is a great number of perceptual hashing techniques designed by researches. For example one group of algorithms adapt DCT because of its good signal decorrelation capabilities. This allows to extract only useful, describing an image content information. A DCT-based robust to JPEG compression image authentication scheme is presented by Lin and Chang in [2]. They use invariant relationships between DCT coefficients in different image blocks. Another perceptual image hashing method called pHash is introduced by Zauner in [3] and is being widely used nowadays due to its effectiveness and simplicity. It uses DCT followed by features reduction, selection of low frequencies and averaging. Similarly, Weng and Preneel utilise DFT to extract image features and secure their perceptual hash with a key [4] which is an essential part of protecting algorithm from attacks.

Zhang and Yang calculate perceptual hash using the weighted average of the Hu invariant moments [5]. The purpose of such approach is to detect original image mirroring if any. Another method similar to the one proposed by Monga and Evans is based on the detection of key-points which are robust to non-malicious manipulations [6]. The general drawback of such methods is the significant sensitivity to both global and local distortions.

Later, new perceptual image hashing techniques called average hashing (aHash) and difference hashing (dHash) are considered by Krawetz [7]. According to the presented results, dHash outperforms aHash and pHash algorithms in terms of the robustness to content-preserving operations. Kozat and Venkatesan use spectral matrix invariants determined by singular value decomposition to be detected even after some part of image is cropped [8]. Lefebvre and Macq present *Rash* a scheme which is based on largely used Radon transform [9]. The algorithm approaches strong tolerance to image rotation while being still able to detect image deformations. Tang and Zhang utilize ring partitioning scheme and non-negative matrix factorization for the same purpose [10]. The method of perceptual image hashing by means of a three–level DWT which uses Daubechies wavelets and iterative geometric operations is suggested by Mehmet, Kivanc Mihcak and Ramarathnam Venkatesan in [11]. Similar effective wavelet-based algorithms are gaining ground under the name of wHash [12].

Considering all the already presented prominent advantages of the perceptual hash algorithms, it is clearly seen that there still is a gap in the research of tolerance to significant brightness and contrast adjustments. The general problem belongs to the fact that these operations can lead to pixel value overflows and thus to original information loss. In order to overcome this issue a novel algorithm of cumulative histogram slicing is suggested. The technique can be used in conjunction with the already existing perceptual image hashing methods as an additional layer of the tolerance to brightness and contrast corrections.

III. Proposed perceptual hashing scheme

It was already stated that our scheme can be used as a base for the already existing perceptual image hashing methods. That is why we split the proposed scheme into two separate steps III-A and III-B as follows.

A. Cumulative histogram slicing step

In order to make perceptual image hashing algorithm robust to brightness and contrast adjustments some invariant image characteristics should be extracted. The following mathematical background is involved in these operations.

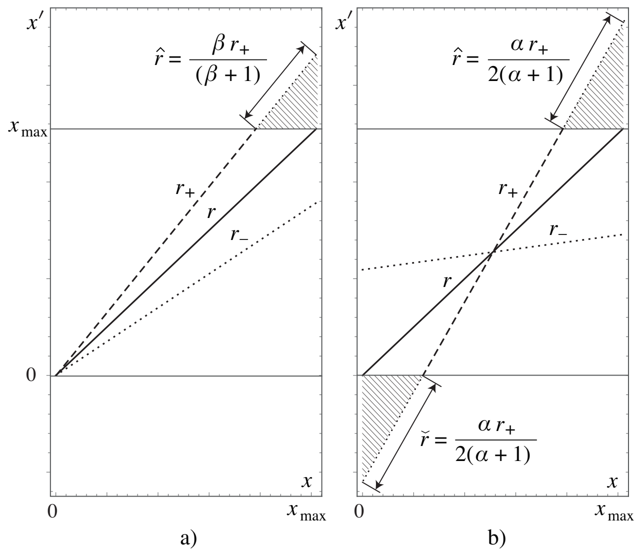

Fig. 1. An effect of brightness (a) and contrast (b) adjustments on output pixel values x' given by source x. The highlighted areas are cut after operations performing

1) Brightness adjustment: Let $X = (x_{m,n}) \in \mathbb{R}^{N \times N}$ be a matrix of pixel values of the source square image with dimensions N^2 for which perceptual hash is going to be obtained. Brightness is a visual perception measure which describes the visible image content illumination. Global brightness correction is usually assumed to be a linear transformation of the matrix X namely

$$X' = X\,(1 + \beta)\,,$$

where $X' = \left(x'_{m,n}\right) \in \mathbb{R}^{N \times N}$ is a matrix of the pixel brightness values before and after transform respectively, $\beta \in \mathbb{R}$ is a brightness adjustment parameter. The brightness of every pixel increases if $\beta > 0$, stays constant if $\beta = 0$ and decreases otherwise.

2) Contrast adjustment: Unlike the brightness correction, contrast adjustment can make one pixel values higher while reducing the other ones. The purpose is to make light and dark image areas more distinguishable. The output is computed as

$$X' = (1 + \alpha)\left(X - \frac{r_{\max}}{2}\right) + \frac{r_{\max}}{2},$$

where $\alpha \in \mathbb{R}$ is the contrast adjustment parameter; $r_{\max}$ is the maximum image pixel value allowed.

In reality the digital image processing implies the limits on the lowest and highest pixel values. Usually one deals with the eight-bit raster images which means that no pixel can be higher than $2^8 - 1 = 255$ or lower than 0. This is why image processing tools crop out the overflowing values which leads to some part of the image histogram being cut. An example of brightness adjustments effect with $\beta = -0.3$ and $\beta = 0.3$ on output pixel values x' by given source values x and dynamic range r is presented in Fig. 1-a. Consider how part of the transformed dynamic range r_+ is cut in the later case. The length of the cut histogram tail $\hat{r}$ can be at most $\beta r_+ / \left(\beta + 1\right)$. A contrast adjustment can lead to overflows in both upper

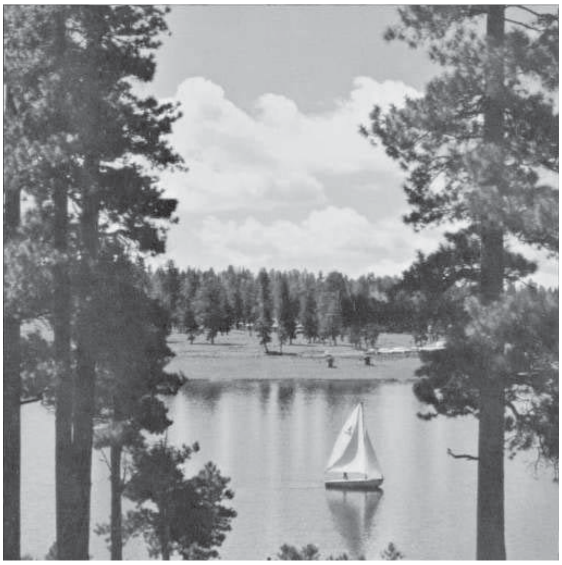

Fig. 2. An example test image “Sailboat on lake”

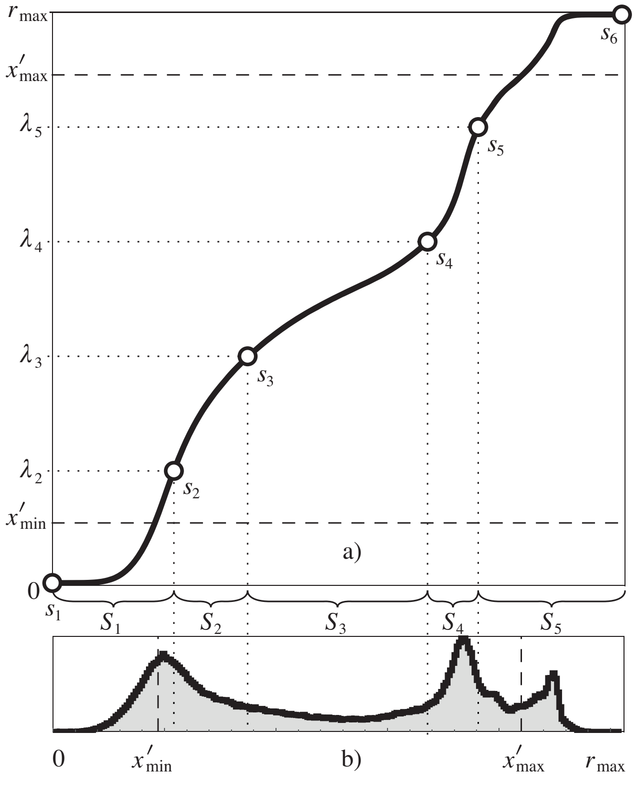

Fig. 3. Cumulative (a) and ordinary (b) histograms mapping scheme given by $\Lambda = 5$ calculated for the test image “Sailboat on lake”

and lower limits. Fig. 1-b clearly shows that this adjustment approaches the same maximum relative histogram cut length $\hat{r} = \check{r} = \alpha r_+ \; 2\left(\alpha + 1\right) = \beta r_+ / \left(\beta + 1\right)$ when

$$\alpha = \frac{2\beta}{1 - \beta}. \tag{1}$$

The ordinary image histogram m_i is a discrete function of the pixel value $x \in [0, r_{\max}]$ which counts the observed number of image pixels whose values equal to x. The cumulative image histogram M_i of the ordinary histogram m_i is defined as

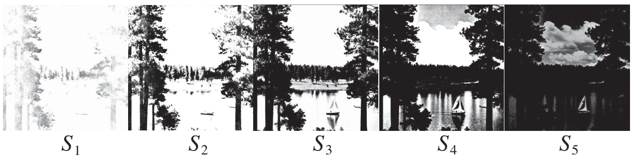

Fig. 4. Images given by truncating of the original image histogram at points λ_i correspondent to cumulative histogram slices S_i for $\Lambda = 5$

$$M_i = \sum_{j=1}^{i} m_i.$$

Let us consider an example image “Sailboat on lake” from “Miscellaneous” volume of the SIPI database [14]. The image is shown in Fig. 2. Its ordinary and cumulative histograms are presented in Fig. 3. We define cumulative histogram slicing points $\lambda_i \in \mathbb{R}^{\geqslant 0}$ as

$$\lambda_i = \frac{i r_{\max}}{\Lambda}, \quad i = 0, 1, \ldots, \Lambda,$$

where Λ is a number of the resulting slices. Then we map $\Lambda + 1$ points λ_i to the correspondent points s_i lying on the ordinary histogram. Points s_i are used to calculate Λ image slices S_i by means of cutting out image histogram at these points. As a result we achieve the Λ slices as it is shown in Fig. 4.

Every slice S_i describes a separate layer of image pixel values and is considered to be robust to the image adjustments except for ones which overflow $x'_{\max}$ or fall below $x'_{\min}$ values and are cut due to truncation procedure. In the latter case there is always a way to check if the cumulative histogram contains the specific slice S_i by looking up the correspondent point λ_i. Otherwise the slice is considered to be unavailable. Such an arrangement makes the proposed system blind. For the algorithm itself it makes no sense if the slice is unavailable due to the performed adjustments or because it was not present in the image at all. For example we can see that after some contrast adjustment the histogram is within the new range from $x'_{\min}$ to $x'_{\max}$. The slices S_1 and S_5 are not feasible.

B. Perceptual image hashing calculation step

The next point to consider is a calculation of image features with the accomplished slices S_i given. Keep in mind that this step of the suggested perceptual image hashing algorithm can be replaced by any other existing one. We apply the HWT transform here just to demonstrate the minimal working solution. In this way the obtained image slices are used to calculate final hashes.

Let $F = \{F_i : \forall F_i, i \in \mathbb{Z}^+, i \leqslant \Lambda\}$ be a set of calculated perceptual hashes for the image X such as

$$F_i = \begin{cases} \operatorname{vec}\left(\Delta \left[\frac{Y_i}{\Delta}\right]\right), & \lambda_i \leqslant x_{\max},\ \lambda_{i+1} \geqslant x_{\min}, \\ \varnothing & \text{otherwise}, \end{cases} \tag{2}$$

where $\Delta \in \mathbb{R}^+$ is a quantization parameter, $\operatorname{vec}(A)$ is a row-by-row vectorization operation applied to the matrix A, $[.]$ is a rounding operation, Y_i is a LL4 coefficient matrix sub-band

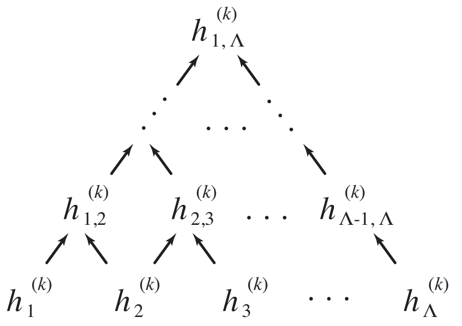

Fig. 5. The suggested hash graph structure to be used with LSH algorithm in the fast image retrieval scheme

of the discrete four-level Haar wavelet transform performed over the slice S_i. Here the LL4 sub-band is used because of its known robustness to the local image noise which can be introduced due to various content-preserving operations. The forward Haar wavelet transform can be calculated as follows

$$Y_i^{(l)} = H_l S_i H_l^{\intercal},$$

where l denotes the level of HWT, $Y_i^{(l)}$ is a matrix of HWT coefficients at level l, $H_l^{\intercal}$ is a transposed version of Haar matrix H_l which can be obtained using the recurrent relation [15]

$$H_0 = [1], \quad H_l = \frac{1}{\sqrt{2}} \begin{bmatrix} H_{l-1} & H_{l-1} \\ H_{l-1} & -H_{l-1}, \end{bmatrix}, \quad l \in \mathbb{Z}^+,$$

and it determines the $(2^l \times 2^l)$-Haar single level matrices H_l. The next level l of HWT can be obtained if the $(2^{l-1} \times 2^{l-1})$-submatrix of HWT approximation coefficients is used instead of the original S_i.

Another aspect of the proposed scheme is that the value of the quantization parameter Δ has to be chosen as a trade-off between perceptual hashing robustness to content-preserving operations and sensitivity to image tampering. For the two images – X and X' – we define an image content equality rule as

$$\begin{matrix} X \text{ and } X' \\ \text{are equal} \end{matrix} \iff \forall i : \max_j |F_i'(j) - F_i(j)| \leqslant 1 \vee F_i = \varnothing, \tag{3}$$

where $i, j \in \mathbb{Z}^+$, $i \leqslant \Lambda$, $j \leqslant N^2/2^{4\times 2}$, $F_i\,(j)$ and $F_i'\,(j)$ are the j-th elements of F_i and F_i' perceptual hashes correspondingly.

Such an equality measure is chosen due to the following considerations. Firstly, there should be an option to use a strong cryptographic digital signature for the image content authentication applications. It can be achieved by means of a well-known three-bit quantization technique [16]. The scheme allows one to apply conventional cryptographic hash function to each of the F_i if condition (3) is fulfilled. Secondly, one would have an ability to design a fast database image retrieval system by the given content query. For this purpose the probabilistic technique such as LSH is being actively used nowadays [13]. There are many LSH algorithms which assume that data are hashed to the same "buckets" with some predefined probability. Taking into account that the suggested perceptual image hashing method uses up to Λ separate hashes in the set F, an adapted LSH based on stable distributions can be utilized. For instance an applied to some vector v LSH algorithm produces K cryptographic hashes h_k, $k \in \mathbb{Z}^+$, $k \leqslant K$. Then h_k are stored using the prebuilt K independent hash tables in the database. For the details on the collision probability which can be given by LSH family refer to [13]. It is more important that LSH achieves constant time nearest neighbours querying.

In order to apply LSH technique to the set F we suggest to use the graph structure. The root hashes $h^{(k)}$ are calculated with use of the k-th LSH family and are presented in Fig. 5. The rest of the vertices ${h^{(k)}}_{p,q}$ are obtained by hashing two concatenated neighbouring parents. The total amount $\mathcal{L}$ of involved hashes are given by

$$\mathcal{L} = \frac{1}{2} K\Lambda(1 + \Lambda). \tag{4}$$

We assume images X and X' to have the same image content except for the possible brightness and contrast corrections if at least one of the $\mathcal{L}$ calculated hashes matches. The scheme gracefully handles the situation when some of the slices are undefined under the condition in (2). It should be noted that slices at the boundaries of F can escape due to the nature of brightness and contrast correction operations described above. The cost for the near-constant time retrieval is the much higher memory consumption. However this is still reasonable in cases of huge image sets.

IV. Experiments on the proposed perceptual image hashing algorithm

In the proposed paper we mostly focus ourselves on the ability of our method to be tolerant to brightness and contrast adjustment manipulations and to be able to detect image tampering. It is necessary to investigate the acceptable values of the parameter Δ which is responsible for the sensitivity of the proposed system. The test image database from SIPI [14] which contains 38 images including natural images, people, aerial, monochrome and other is used. Note that the images from this set have various levels of global luminosity and are picked up by SIPI in a way to be entirely different from each other.

Performed experiments have shown that the parameter $\Lambda = 5$ is good enough for the evaluated SIPI image set. However, for the images with the more wide dynamic range such as the medical images or the ones which have very narrow cumulative histogram function we recommend one to male the Λ parameter be greater in order to capture more image slices. On the other hand, the more Λ is, the more hashes are involved according to (4) and the memory consumption tends to increase. In case of 512×512 image and $\Lambda = 5$ the whole perceptual image hash (2) requires up to $5 \times 2^{4\times 2} = 1280$ elements. If one uses a three-bit quantization technique then it will have $1280 \times 3 = 3840$ bits to store the perturbation bits as well as 512 bits for the strong ECDSA signature [17]. This amount of memory is practically acceptable for the digital image watermarking schemes like DWT-based embedding. However, it can still be reduced by adjusting the parameters of the proposed method.

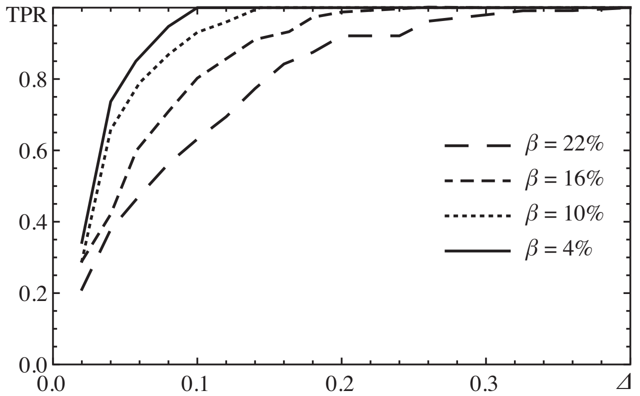

Fig. 6. Robustness to brightness increase operation by factor $\beta \in \{4\%, 10\%, 16\%, 22\%\}$ as a dependence of TPR against a quantization step Δ

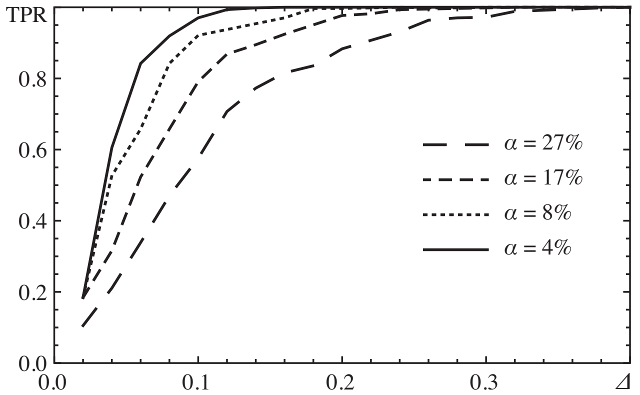

Fig. 7. Robustness to contrast increase operation by factor $\alpha \in \{4\%, 8\%, 17\%, 27\%\}$ as a dependence of TPR on a quantization step Δ

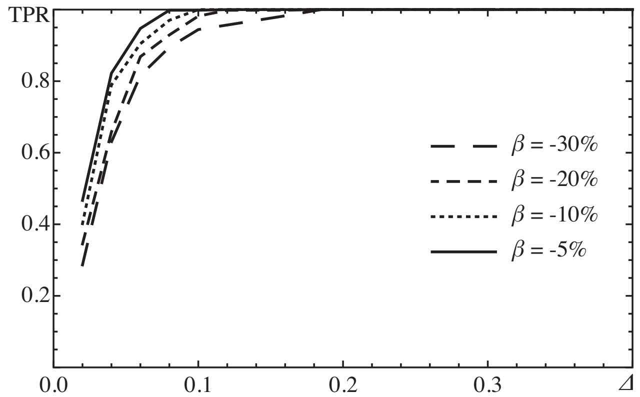

Fig. 8. Robustness to brightness decrease operation by factor $\beta \in \{-5\%, -10\%, -20\%, -30\%\}$ as a dependence of TPR on quantization step Δ

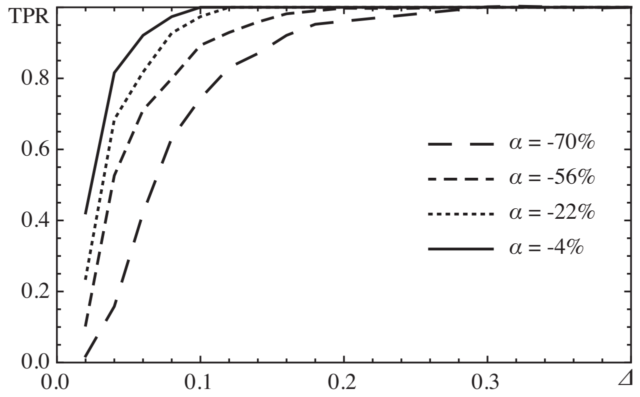

Fig. 9. Robustness to contrast decrease operation by factor $\alpha \in \{-4\%, -22\%, -56\%, -70\%\}$ as a dependence of TPR on quantization step Δ

A. Tolerance to the image contrast and brightness corrections

To verify the robustness our experiments are arranged in the following way. First, we populate the test image set on which content-preserving manipulations are performed. Brightness of each test image is adjusted according to $\beta = i/50$, $i \in \mathbb{Z}$, $-15 \leqslant i \leqslant 15$. Contrast correction is applied with the parameters α equivalent to the brightness parameter β in agreement with (1). Secondly, the suggested perceptual image hashing algorithm is applied to the test image set sequentially. Different values of the parameter Δ are applied in (2) all over the set.

As for the image content authentication purposes it is important to test the suggested method under the rule (3). Calculated perceptual image hashes are compared with those obtained from the original versions of the images. We define True Positive Rate (TPR) as the ratio of the cases when algorithm successfully classified the adjusted image versions as authentic ones comparing with their originals.

The dependencies of TPR on quantization parameter Δ under the brightness increase operation with various parameters β are shown in Fig. 6. The plot is compared with the similar dependencies calculated for the contrast increase operation with the different parameter α in Fig. 7. It is clearly seen that the suggested algorithm has a bit better tolerance to the contrast increase operation with the given absolute values of parameters α and β. This also has been previously confirmed by Fig. 1 and relation (1). Note that we compare brightness and contrast increase operations to each other first because both can lead to the histogram cut due to the truncation.

On the other hand, robustness to brightness and contrast decrease operations with various parameters α and β as the dependencies of TPR on quantization parameter Δ are presented in Fig. 8 and 9 correspondingly. As we have already mentioned, the brightness and contrast decrease operations just shrink the histogram. An information loss occurs only because of the histogram rescaling to the narrower boundaries. Thus it is what the suggested algorithm should have better tolerance to. This statement as well as the relation (1) are fully confirmed by the plots. To illustrate the tolerance of the suggested algorithm to the contrast decrease operation observe how significantly the image is changed even with parameter $\alpha = -50\%$ in Fig. 10 comparing to Fig. 2.

B. Sensitivity to the image tampering

The most important ability of any content image authentication system is to detect image tampering. In order to validate how the suggested system is sensitive to such manipulations we have arranged the following experiment.

All of the 38 SIPI images are tampered within the randomly chosen circle areas of radius ρ. Each of the tampering areas

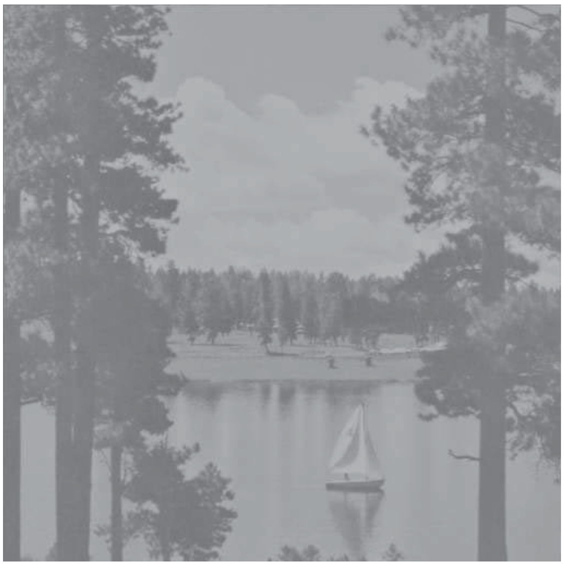

Fig. 10. An example of contrast decrease operation with parameter $\alpha = -50\%$ applied to the test image "Sailboat on lake"

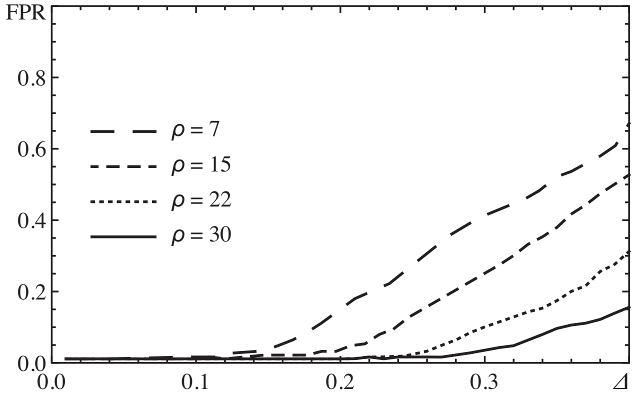

Fig. 11. Sensitivity to image tampering within radius $\rho \in \{15,\ 30,\ 45, 60\}$ as a dependence of FPR on quantization step Δ

is filled up with the randomly chosen pixel values within the image dynamic range. We repeat such tampering ten times for the each of the images in the test set. Similarly to TPR measure we define False Positive Rate (FPR) as the ratio (define fraction here) of cases when the suggested algorithm classified the tampered image as the authentic one.

The dependencies of FPR on quantization parameter Δ for different tampering radius ρ are presented in Fig. 11. One can see the more tampering radius ρ is, the less FPR given the same values of parameter Δ, as expected. Anyway, the randomly chosen tampering algorithm is not such good. Sometimes this makes the algorithm totally fused in the tampering area with the original image content. However, such an experimental technique allows us to see the discriminative ability of the suggested method in general.

C. Comparison with the existing methods

Although the general performance of the proposed method has been investigated, an essential evaluation has to be done among the other already existing perceptual image hashing techniques. We chose a few popular methods with different kind of features: pHash [3], aHash [7], dHash [7] and wHash [12]. The previously obtained in Sections IV-A, IV-B images with content-preserving operations and tampering performed are involved into the following experiment. For the proposed method $\Delta = 0.12$ and $\Lambda = 5$ have been selected while the other algorithms were tested with their default hyperparameters. The overall perceptual image hash length is set to 1024 for all methods. The comparison in terms of robustness to brightness and contrast corrections as the dependencies of TPR on parameters α and β are presented in Fig. 12 and Fig. 13 correspondingly. On the other hand, we compare the sensitivity to image tampering in terms of True Negative Rate (TNR $= 1-$FPR) as the dependencies on radius ρ in Fig. 14. One can confirm that the proposed algorithm has mostly the best robustness to brightness and contrast changing operations and achieves sensitivity to image tampering at the same time. The discriminative ability is close to the best of pHash algorithm.

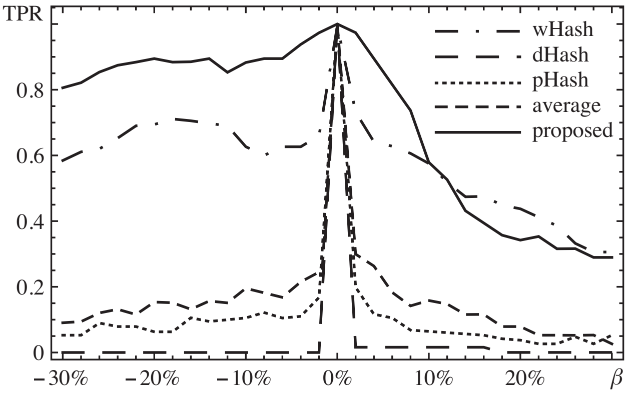

Fig. 12. The comparison of the existing perceptual image hashing methods with the proposed one in terms of robustness to contrast correction as the dependencies of TPR on parameter β obtained by $\Delta = 0.12$

As a final remark, one would have to choose lower values for the parameter Δ when the proposed method is used in a content image authentication application. Besides, the case of the false alarm can have much less fatal consequences than just missed duplicated content. On the other hand, it is better to select higher values for Δ not to miss any duplicate when image retrieval application is used.

V. Conclusion

It is shown that the proposed algorithm can be utilized in various practical applications. For instance image content authentication scheme can verify if there were only content-preserving operations performed and it can detect image tampering, if any. This is done with the help of the well known three-bit quantization technique which requires the more extensive memory use through. Another practical application of the proposed method is a constant time duplicate image retrieval from the database. The latter requires to apply LSH to each of the obtained perceptual image hashes and then to construct the proposed hash graph.

According to the performed experimental results we can conclude that the perceptual image hashing algorithm has good

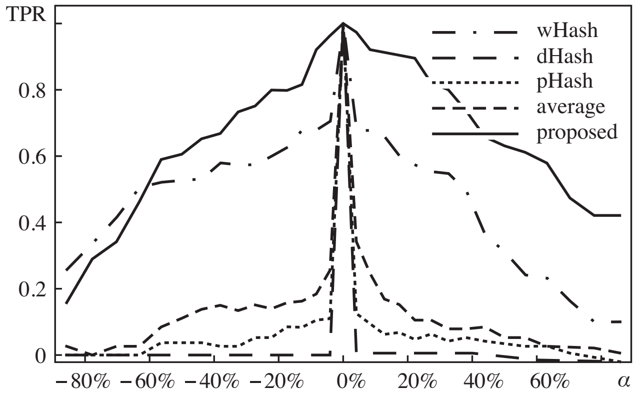

Fig. 13. The comparison of the existing perceptual image hashing methods with the proposed one in terms of robustness to brightness correction as the dependencies of TPR on parameter α obtained by $\Delta = 0.12$

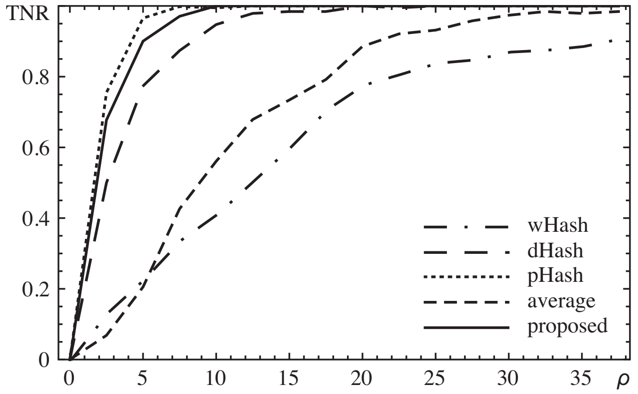

Fig. 14. The comparison of the existing perceptual image hashing methods with the proposed one in terms of sensitivity to image tampering as the dependencies of TPR on radius ρ obtained by $\Delta = 0.12$

robustness to both significant brightness and contrast corrections. The introduced algorithm has shown the best results in terms of robustness to brightness and contrast corrections among the other popular perceptual image hashing techniques. However, it has better tolerance to the contrast and brightness increase operations comparing with the decrease ones given the same parameters α and β. This follows the fact that image histogram can be truncated after increase operations and it has been confirmed by the experimental results. We recommend to choose quantization parameter $\Delta \approx 0.12$ as a good trade-off which approaches TPR $= 1$ for the contrast and brightness corrections up to $15\% - 20\%$ and still has an ability to detect image tampering within radius $\rho \approx 22$. It became possible due to the use of the proposed cumulative histogram slicing technique as an initial step. We use a simple four-level HWT-based perceptual image hashing scheme over this step for illustrative purposes only. Moreover the suggested histogram slicing step can easily be adapted to be a base for the other more sophisticated perceptual hashing schemes in order to add an additional layer of tolerance to both brightness and contrast adjustments.

We conclude that the above mentioned facts qualify the proposed algorithm to be one which can be effectively used in some practical applications. However, our scheme can have a higher memory consumption as the drawback. This and robustness to the more wide set of content-preserving operations such as JPEG compression are the points for the further investigation.

References

[1] Cisco, *Cisco Visual Networking Index: Forecast and Trends, 2017–2022*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.pdf, San Jose, 2019.

[2] C. Y. Lin, S.F. Chang, *A robust image authentication method distinguishing jpeg compression from malicious manipulation*, IEEE Transactions on Circuits and Systems for Video Technology 11(2): 153–168. 2001.

[3] C. Zauner, *Implementation and Benchmarking of Perceptual Image Hash Functions*, Master's thesis, Upper Austria University of Applied Sciences, Hagenberg Campus, 2010.

[4] L. Weng, B. Preneel, *A secure perceptual hash algorithm for image content authentication*. In Proceedings of the 12th IFIP TC 6/TC 11 international conference on Communications and multimedia security (CMS'11), Bart De Decker, Jorn Lapon, Vincent Naessens, and Andreas Uhl (Eds.). Springer-Verlag, Berlin, Heidelberg, 108-121, 2011.

[5] B. Zhang, Y. Xin and X.-X. Niu, *Image perceptual hash algorithm based on target character*, 2011 IEEE 13th International Conference on Communication Technology, Jinan, pp. 397-401, 2011.

[6] V. Monga, B.L. Evans, *Perceptual image hashing via feature points: Performance evaluation and trade-offs*, IEEE Transactions on Image Processing 15(11), 3452–3465, 2006.

[7] N. Krawetz, *Kind of Like That*, [Online]. Available: http://www.hackerfactor.com/blog/index.php?/archives/529-Kind-of-Like-That.html, January 2013.

[8] S.S. Kozat, R. Venkatesan and M.K. Mihcak, *Robust perceptual image hashing via matrix invariants*, 2004 International Conference on Image Processing, ICIP 04., Singapore, pp. 3443-3446 Vol. 5. 2004.

[9] F. Lefebvre, B. Macq, J.D. Legat, *Rash: Radon soft hash algorithm*, Proceedings of the European Signal Processing Conference (EUSIPCO'02), Toulouse, France. 2002.

[10] Z. Tang, X. Zhang, S. Zhang, *Robust perceptual image hashing based on ring partition and NMF*, IEEE Trans. Knowl. Data Eng., vol. 26, no. 3, pp. 711-724, Mar. 2014.

[11] M.M. Kivanc and R. Venkatesan, *New Iterative Geometric Methods for Robust Perceptual Image Hashing*, Digital Rights Management Workshop. 2001.

[12] A. Geetanjali, *Aneka- Detecting various forms of the same Wavelet Image Hashing Algorithm*, Master's thesis, California State University Channel Islands, USA, December 2018.

[13] M. Datar, N. Immorlica, P. Indyk, and V.S. Mirrokni, *Locality-sensitive hashing scheme based on p-stable distributions*, In Proceedings of the Twentieth Annual Symposium on Computational Geometry, SCG '04, pages 253-262, New York, NY, USA, 2004.

[14] SIPI, *The USC-SIPI Image Database*, [Online]. Available: http://sipi.usc.edu/database. [Accessed Jul.10, 2019].

[15] P. Porwik and A. Lisowska, *The Haar wavelet transform in digital image processing: its status and achievements*, Int. Journal Machine Graphics and Vision., vol. 13, no. 1, pp. 79–98, 2004.

[16] F. Ahmed and M. Y. Siyal, *A Robust and Secure Signature Scheme for Video Authentication*, IEEE, International Conference on Multimedia and Expo, 2007.

[17] D. Hankerson, A.J. Menezes, S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer, Heidelberg. 2004.