Could DCT Reveal Photorealistic Images?

Konstantinos Annousakis-Giannakopoulos[1], Dimitris Ampeliotis[2], Athanassios Skodras[1]
[1]*Department of Electrical and Computer Engineering*, *University of Patras*, Patras, Greece.
[2]*Department of Computer Engineering and Informatics*, *University of Patras*, Patras, Greece.
E-mails: kannousakisg@gmail.com, ampeliot@ceid.upatras.gr, skodras@upatras.gr

***Abstract*—With the development of computer graphics rendering software, it has become extremely difficult to distinguish whether an image is computer generated or a natural one. Therefore, it is really important to device robust methods for correctly classifying these two categories of images. In this work, a new approach to face the above problem is developed that is based upon the discrete cosine transform (DCT) of an image, in the YCbCr color space. The statistical features extracted, have been tested in suitable databases and the remarkable results indicate that the proposed model has a great potential to be used in digital images forensics.**

***Index Terms*—Computer Graphics, Fake Photos, Digital forensics, Photorealism, Discrete Cosine Transform, DCT, Quantization.**

I. Introduction

Modern techniques, like ray tracing [1], and the rapid development of computer graphics rendering software, like Maya [2] or 3Ds Max [3], allow us to generate complex scenes and photorealistic images. Sometimes, it is impossible even for a naked eye to discriminate if they are representation of a real scene or not [4]. On one hand, the high level of photorealism of computer generated (CG) images, permits their use in various fields, including entertainment, medical sciences, education and industry. On the other hand, due to the fact that image nowadays is used as an evidence of truth in law courts and journalism, it becomes an urgent problem to discriminate CG from natural images, in order to ensure that photographs (PH) will regain their integrity and reliability. For this reason, the validation of authenticity of the images is an important challenge in forensics [5].

However, even if the CGI rendering algorithms succeed a high level of photorealism, it is difficult to create the subtle details that exist in reality. Certain of the attributes that CG have, are the patches of uniform color, subtle color variation, simple scenes and a small number of objects. In addition, it is quite usual to contain additive noise to increase the levels of photorealism.

In this paper, we consider the discrimination of the two image categories, as a passive-blind problem of image forensics [6]. This means, that the classification of an image is not relied on pre-registration information. We describe a statistical method that has its roots in JPEG compression, i.e. it utilizes the DCT [7] of the 8×8 non-overlapping blocks of an image in the YCbCr color space. The features used by the proposed method are the first four statistical moments (mean, variance, skewness and kurtosis) of two sets of error signals, where the first set contains prediction errors for various DCT coefficients, while the second set considers errors in the spatial domain, at different compression scales. Experimental results have shown that the proposed approach achieves an average classification accuracy of 93.14%, for the dataset of [8].

The rest of this work is organized as follows: In Section II the related work is presented. The proposed method is detailed in Section III and experimental results are given in Section IV. Finally, conclusions are drawn in Section V.

II. Related Work

The problem of discriminating CG from PH images constitutes an important topic in digital forensics, and it has drawn significant interest by many researchers. Based upon the kind of features utilised, the various developed methods can, in general, be classified into the following three categories:

A. Methods based on statistical features

In 2005, Farid and Lyu [9] proposed a wavelet-based method. After the construction of a four-scale, three-orientation quadrature mirror filter (QMF) pyramid, the first four order statistics of the wavelet subband coefficient histograms and of the prediction errors, are collected, forming a feature vector of 216 dimensions. For a false-negative rate of 1.2%, the method achieved an accuracy of 66.8% for the detection of natural images. In 2012, Fan et al. [10], proposed the use of a four level contourlet transform in the HSV color space, and considered feature vectors of 384 dimensions. The average accuracy of the method was evaluated at 93.51%.

B. Methods based on geometric features

In 2005 Ng et al. [11], proposed a model based on the different geometric properties of each kind of image and their different generation processes. At the finest scale, they describe the geometry of the scene by a fractal geometry, while at the intermediate scale, differential geometry is used. The proposed method was shown to achieve an average accuracy of 83.5%, utilizing feature vectors of 192 dimensions.

C. Methods based on physical features

The acquisition of photographs is a multi-stage process that includes lens transformation, color filter array (CFA), white balancing, gamma correction and noise addition [12]. Many researchers have focused on detecting the presence (or absence) of traces of the photo generation process in a given image, in order to discriminate PH from CG images. Ng

978-1-7281-4959-2/19/$31.00

et al. [11], studied certain physical differences between the two categories. In 2014, Peng and Zhou [13], tried to find a solution for the problem of the discrimination, based on the impact of CFA interpolation to the local correlation of the so-called photo response non-uniformity (PRNU). Feature vectors, with 9 dimensions, were proposed, utilizing the local variance of various histograms of the PRNU. An identification accuracy of 99.43% was achieved. Furthermore, in 2005, A.C. Gallagher achieved an accuracy of 98.4% by detecting traces of democaising [14].

III. The proposed method

Consider a color image $\mathbf{I} \in \mathbb{R}^{N\times K\times 3}$, where N is the number of rows, K is the number of columns while the third dimension refers to the color components for red, green and blue. Image $\mathbf{I}$ is transformed into the YCbCr color space, and let $\mathbf{J} \in \mathbb{R}^{N\times K\times 3}$ denote the transformed image, where, now, the third dimension refers to the so-called luma, blue-difference and red-difference components. Simulation results not included in this manuscript, showed that the proposed method that utilizes this transformation, leads to increased discrimination accuracy as compared to the approach that does not perform this color transformation. In the following, the process used to compute the statistical features is described in terms of an input image $\mathbf{S} \in \mathbb{R}^{N\times K}$, where $\mathbf{S}$ is used for any of the three spatial components of $\mathbf{J}$, and the process is identical for all components.

Image $\mathbf{S}$ is decomposed into non-overlapping blocks of size 8×8 pixels, denoted as $\mathbf{S}_{i,j}$, where $i = 1, 2, \ldots, I = N/8$ and $j = 1, 2, \ldots, J = K/8$ denote the block indexes, and N, K are assumed to be multiples of 8. The two-dimensional DCT of each spatial block $\mathbf{S}_{i,j}$ is computed, and the resulting DCT block is denoted as $\mathbf{D}_{i,j}$. Furthermore, we denote the DC coefficient in $\mathbf{D}_{i,j}$ as $DC_{i,j}$ and the rest coefficients as $AC_{i,j}^{mn}$, where m and n denote indexes of a coefficient, assuming integer values from 0 to 7. Fig. 1 illustrates the notation used for the DCT coefficients contained in the 8×8 block $\mathbf{D}_{i,j}$.

As also mentioned in the previous section, the proposed approach utilizes two sets of features. The first set of features is collected by working in the DCT domain, where nine linear predictors are used for various DCT coefficients, the prediction errors are extracted and moments of the error signals are computed as features. The second set of features is collected by working in the spatial domain, at different compression scales, where a predictor is used for quantized DC values and moments of the errors are collected as features. In the following, we describe the computation of these two sets of features in greater detail.

A. Features in DCT Domain(First set of features)

For the first set of features, linear predictors for the DC and certain AC coefficients, are computed. Depending on what each predicted coefficient represents, a different set of inputs is considered in the corresponding predictor. In more detail, the choice of inputs for each predictor was based upon the following reasoning:

$DC_{i,j}$	$AC_{i,j}^{01}$	$AC_{i,j}^{02}$	$AC_{i,j}^{03}$	$AC_{i,j}^{04}$	$AC_{i,j}^{05}$	$AC_{i,j}^{06}$	$AC_{i,j}^{07}$
$AC_{i,j}^{10}$	$AC_{i,j}^{11}$	$AC_{i,j}^{12}$	$AC_{i,j}^{13}$	$AC_{i,j}^{14}$	$AC_{i,j}^{15}$	$AC_{i,j}^{16}$	$AC_{i,j}^{17}$
$AC_{i,j}^{20}$	$AC_{i,j}^{21}$	$AC_{i,j}^{22}$	$AC_{i,j}^{23}$	$AC_{i,j}^{24}$	$AC_{i,j}^{25}$	$AC_{i,j}^{26}$	$AC_{i,j}^{27}$
$AC_{i,j}^{30}$	$AC_{i,j}^{31}$	$AC_{i,j}^{32}$	$AC_{i,j}^{33}$	$AC_{i,j}^{34}$	$AC_{i,j}^{35}$	$AC_{i,j}^{36}$	$AC_{i,j}^{37}$
$AC_{i,j}^{40}$	$AC_{i,j}^{41}$	$AC_{i,j}^{42}$	$AC_{i,j}^{43}$	$AC_{i,j}^{44}$	$AC_{i,j}^{45}$	$AC_{i,j}^{46}$	$AC_{i,j}^{47}$
$AC_{i,j}^{50}$	$AC_{i,j}^{51}$	$AC_{i,j}^{52}$	$AC_{i,j}^{53}$	$AC_{i,j}^{54}$	$AC_{i,j}^{55}$	$AC_{i,j}^{56}$	$AC_{i,j}^{57}$
$AC_{i,j}^{60}$	$AC_{i,j}^{61}$	$AC_{i,j}^{62}$	$AC_{i,j}^{63}$	$AC_{i,j}^{64}$	$AC_{i,j}^{65}$	$AC_{i,j}^{66}$	$AC_{i,j}^{67}$
$AC_{i,j}^{70}$	$AC_{i,j}^{71}$	$AC_{i,j}^{72}$	$AC_{i,j}^{73}$	$AC_{i,j}^{74}$	$AC_{i,j}^{75}$	$AC_{i,j}^{76}$	$AC_{i,j}^{77}$

Figure 1: An illustration of the notation used for the DCT coefficients in block $\mathbf{D}_{i,j}$

- For the prediction of the DC coefficient $DC_{i,j}$ of the block $\mathbf{D}_{i,j}$, which represents the average value of the respective spatial block $\mathbf{S}_{i,j}$, the predictor utilizes as inputs the DC coefficients of neighbouring blocks, considering an 8×8 (block) neighbourhood.
- For the prediction of the coefficients $AC_{i,j}^{01}$, $AC_{i,j}^{02}$ and $AC_{i,j}^{03}$, that represent horizontal variations, the DC coefficients of adjacent blocks along the same (block) row, namely $DC_{i,j-1}$ and $DC_{i,j+1}$, are considered as inputs for the respective predictors.
- Similarly, for the prediction of the coefficients $AC_{i,j}^{10}$, $AC_{i,j}^{20}$ and $AC_{i,j}^{30}$, that represent vertical variations, the DC coefficients of adjacent blocks along the same (block) column, namely $DC_{i-1,j}$ and $DC_{i+1,j}$, are considered as inputs for the respective predictors.
- For the prediction of the coefficients $AC_{i,j}^{11}$ and $AC_{i,j}^{21}$, that represent diagonal variations, the DC coefficients of diagonally adjacent blocks, namely $DC_{i-1,j-1}$, $DC_{i-1,j+1}$, $DC_{i+1,j-1}$ and $DC_{i+1,j+1}$, are considered.

AC coefficients further away from the DC term of the block are not used, since their values are very small and do not provide any important additional information for the problem at hand. Based upon the previous, the following nine linear predictors are considered:

$$\begin{aligned} DC_{i,j} \approx\ & w_1 DC_{i-1,j-1} + w_2 DC_{i-1,j} \\ & + w_3 DC_{i-1,j+1} + w_4 DC_{i,j-1} \\ & + w_5 DC_{i,j+1} + w_6 DC_{i+1,j-1} \\ & + w_7 DC_{i+1,j} + w_8 DC_{i+1,j+1}\ , \end{aligned} \tag{1}$$

$$AC_{i,j}^{01} \approx w_9 DC_{i,j-1} + w_{10} DC_{i,j+1}\ , \tag{2}$$

$$AC_{i,j}^{02} \approx w_{11} DC_{i,j-1} + w_{12} DC_{i,j+1}\ , \tag{3}$$

$$AC_{i,j}^{03} \approx w_{13} DC_{i,j-1} + w_{14} DC_{i,j+1}\ , \tag{4}$$

$$AC_{i,j}^{10} \approx w_{15} DC_{i-1,j} + w_{16} DC_{i+1,j}\ , \tag{5}$$

$$AC_{i,j}^{20} \approx w_{17} DC_{i-1,j} + w_{18} DC_{i+1,j}\ , \tag{6}$$

$$AC_{i,j}^{30} \approx w_{19} DC_{i-1,j} + w_{20} DC_{i+1,j} \ , \tag{7}$$

$$AC_{i,j}^{11} \approx w_{21} DC_{i-1,j-1} + w_{22} DC_{i-1,j+1} + w_{23} DC_{i+1,j-1} + w_{24} DC_{i+1,j+1} \ , \tag{8}$$

and

$$AC_{i,j}^{21} \approx w_{25} DC_{i-1,j-1} + w_{26} DC_{i-1,j+1} + w_{27} DC_{i+1,j-1} + w_{28} DC_{i+1,j+1} \ . \tag{9}$$

The weights w_1 to w_{28} in the previous relations are computed using the method of least-squares, separately for each predictor. A procedure similar to that proposed in [9] is followed, reproduced here for giving a complete description of the adopted methodology. In more detail, for each of the above predictors, a linear system of equations is constructed, having the form:

$$\mathbf{v} = \mathbf{Q}\mathbf{w} \ , \tag{10}$$

where the vector $\mathbf{v}$ collects the "desired response" values, that appear on the left-hand side of a predictor relation, for all possible blocks $\mathbf{D}_{i,j}$. Also, matrix $\mathbf{Q}$ has a number of rows equal to that of vector $\mathbf{v}$, a number of columns equal to the number of weights utilised by the corresponding predictor, and collects the inputs (e.g. coefficients from neighbouring blocks) as they appear on the right-hand side of the corresponding predictor relation. Finally, the vector $\mathbf{w}$ contains the (unknown) weights utilized by the predictor. The method of least squares aims to minimize the sum of squared residuals, that is, the cost function $c(\mathbf{w})$ employed is given by

$$c(\mathbf{w}) = \|\mathbf{v} - \mathbf{Q}\mathbf{w}\|_2^2 \ . \tag{11}$$

The minimization is accomplished, by differentiating with respect to $\mathbf{w}$. Afterwards by setting the result equal to zero, we arrive at the following solution:

$$\frac{\vartheta c(\mathbf{w})}{\vartheta \mathbf{w}} = 0 \Rightarrow \mathbf{w}_o = (\mathbf{Q}^T\mathbf{Q})^{-1}\mathbf{Q}^T\mathbf{v} \ , \tag{12}$$

where $\mathbf{w}_o$ denotes the optimal weights vector. For each of the above predictors, we can assume that the inverse of the matrix $\mathbf{Q}^T\mathbf{Q}$ exists. After the computation of the weights, the logarithm of the respective error is computed, for each one of the considered predictors, as shown by

$$\mathbf{e} = \log\left(|\mathbf{v} - \mathbf{Q}\mathbf{w}_o|\right) \ . \tag{13}$$

Finally, the first four moments (mean, variance, kurtosis and skewness) for each of the error vectors are collected. Following the same process for each component (Y, Cb, Cr), independently, 108 different features are computed (36 for each of the three channels).

B. Features in quantized DCT domain, by using information from initial YCbCr color space(Second Set of Features)

Since the set of features considered in the previous utilizes information from the DCT domain of a given image, it was decided to complement that set with some more features computed in the spatial domain. Furthermore, in order to emphasize any possible different behaviour between CG and PH images in terms of compression, several different compression ratios, are utilized.

Consider again an image block $\mathbf{S}_{i,j}$ and the corresponding two-dimensional DCT block $\mathbf{D}_{i,j}$. Similarly to our previous description, these symbols are used for any of the three components of the input image, that undergo a similar processing. Block $\mathbf{D}_{i,j}$ is quantized by employing the respective quantization table of the JPEG standard [15], which is different for each component and for a different quality factor. We utilize the corresponding quantization tables for Y, Cb and Cr, and consider the quality factors 20, 50 and 80. Although this procedure will produce a different result each time, we use the same symbol $\tilde{\mathbf{D}}_{i,j}$ to denote the resulting quantized block. Also, we denote the quantized DC coefficient in the block $\tilde{\mathbf{D}}_{i,j}$ as $\widetilde{DC}_{i,j}$.

In a manner somewhat similar to that followed in the previous, we consider an estimate of the quantized DC coefficient of a block, given by the relation

$$\widehat{\widetilde{DC}}_{i,j} = \frac{1}{2}\left(\widetilde{DC}_{i,j-1} + L_{i,j}\right) + \frac{1}{2}\left(\widetilde{DC}_{i,j+1} + R_{i,j}\right) \tag{14}$$

where

$$L_{i,j} = \frac{1}{8}\sum_{m=0}^{7}\left(S_{i,j}^{m0} - S_{i,j-1}^{m7}\right)$$

and

$$R_{i,j} = \frac{1}{8}\sum_{m=0}^{7}\left(S_{i,j}^{m7} - S_{i,j+1}^{m0}\right)$$

denote the averages of the differences of (spatial) image intensities along the edges of the 8×8 block $\mathbf{S}_{i,j}$ and its (block) row-neighbouring blocks $\mathbf{S}_{i,j-1}$ and $\mathbf{S}_{i,j+1}$, as illustrated in Fig. 2. Also, note that we have used the symbol $S_{i,j}^{mn}$ to denote the value of the original image at the 8×8 block $\mathbf{S}_{i,j}$, where m and n denote the indexes of a pixel in that block, assuming integer values from 0 to 7.

Thus, for each block, we define the error term

$$e_{i,j} = \log\left(\left|\widehat{\widetilde{DC}}_{i,j} - \widetilde{DC}_{i,j}\right|\right) \ ,$$

and, as in the first set of features, the first four moments of the error are collected, which lead us to a second set of 36 features (4 moments, 3 color components, 3 different quality factors). Thus, the complete feature vector contains a total of 144 features, that will be used to solve the problem of

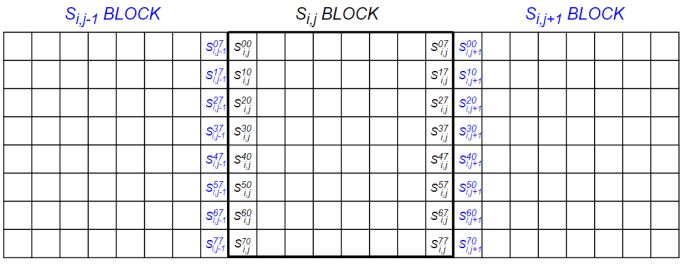

Figure 2: Borders of an 8x8 block in YCbCr color space

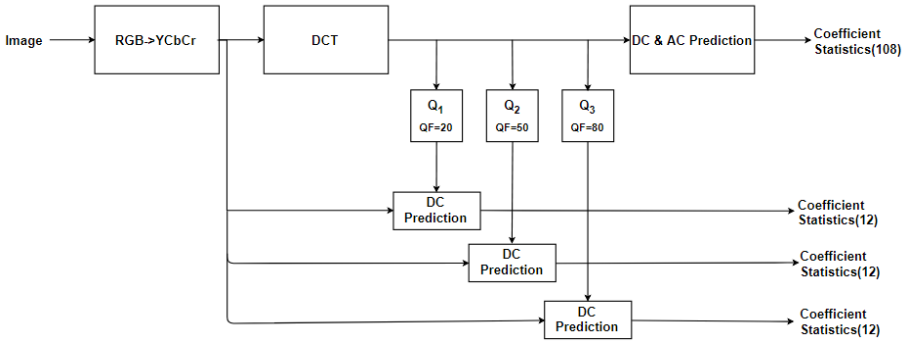

Figure 3: Feature extraction process

the discrimination. The whole process of the feature vector extraction is illustrated in Fig. 3.

IV. Experimental Results

A. Classification

A widely used algorithm for classification in machine learning, is Support Vector Machines (SVM), so for the classification we employed a SVM with radial basis function (RBF) kernel, implemented in LIBSVM [16]. By performing grid search through a five-fold cross validation on the training set, the best classifier parameters are obtained (SVM kernel parameter γ and the regularization parameter C). This means, that firstly, the database of images is partitioned in training and testing set with a ratio of 80% and then, by using the training set, we decided the classifier parameters. Afterwards, the testing set is supplied into LIBSVM, where the class of the images is identified.

B. Experimental Results

In order to evaluate the performance of the proposed approach, numerical experiments using an image database provided by Tokuda et al. [8] were conducted. Based on this image database, that contains 60,000 photographs and 7,000 graphics of various resolutions, a subset of 4,850 PH and 4,850 CG images were selected for our experiments, ensuring that all images have a size of at least 512×512 pixels.

The image database is randomly partitioned in 7,760 images for training (80%) and 1,940 for testing (20%). Also, 10 different random such partitions were examined. Table I summarizes the average accuracy obtained by these experiments.

As seen in Table I, the accuracy of the proposed classifier is 93.14% and therefore outperforms all the individual methods that Tokuda et al. [8], have implemented at the same image database. Note that, our method has good results in both of two classes. In addition, Table I shows the average classification accuracy of the two sets of features independently. Even if the first set provides better classification accuracy, both of them are required for better results.

Another measure which reveals the good performance of our classifier is the Receiver Operator Characteristics (ROC) curve. The Area Under Curve (AUC) of ROC curve is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance [17]. As illustrated in Fig. 4, the AUC of the proposed method is approximately 0.98, as the best classifier of [8].

Table I: Experimental Results for Different Sets of Features

	PH (%)	CG (%)	Accuracy(%)
1^{st} set of features	92.35	91.04	91.70
2^{nd} set of features	89.64	87.02	88.33
Joint set of features	93.59	92.69	93.14

Furthermore, we wondered which of the color channel or combination of two channels, gives us the best accuracy. As we see in Table II, the chroma channels have better classification accuracy than the luminosity one, as a result of non naturality of colors variance across a CG image, in contrast with natural images. From Table I and Table II, the conclusion drawn, is that all the features are important for the construction of the final classifier, in order to have the best possible performance.

In addition, due to the fact that the database that we worked on, contained JPEG compressed images(lossy compression), we wondered if the loss of information during JPEG compression, affects the classification results. To this end, we performed the following experiment. Firstly, we applied the proposed method of discrimination, in a small database of PNG compressed images, that contained the same number of CG and PH images. Afterwards, the same image database has been JPEG compressed, with QF=90% and then our method, applied once again. As expected, the classification accuracy of PNG images was better than the JPEG compressed

Table II: Experimental Results for Different Combinations of Channels

	PH (%)	CG (%)	Accuracy(%)	# of Features
Y	84.25	82.3	83.28	48
Cb	92.32	90.38	91.35	48
Cr	91.76	91.53	91.65	48
Y Cb	92.8	91.8	92.30	96
Y Cr	92.80	91.29	92.05	96
Cb Cr	92.57	91.49	92.03	96
Y Cb Cr	93.59	92.69	93.14	144

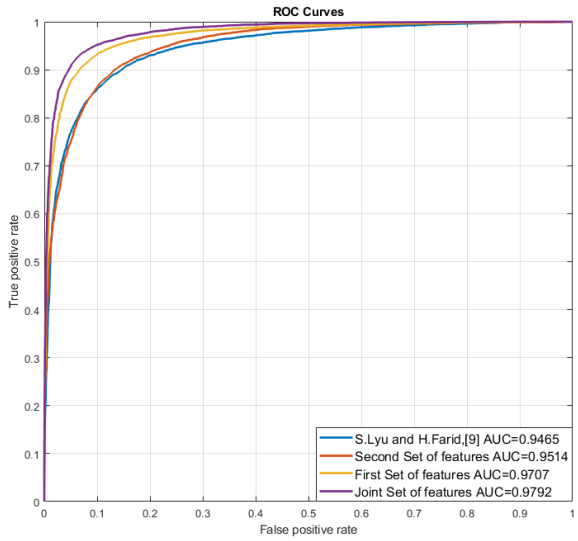

Figure 4: ROC curves for different sets of features and for method [9].

images, as a result of the loss of information during the JPEG compression.

Furthermore, to analyze the influence of the central image block on the identification accuracy, experiments have been performed with different size of the central block(512x512, 256x256 etc). For each reduction of the size of the central block, the results were produced faster but less accurate. This fact indicates that the classification accuracy is positive proportional to the size of the central image block, while the speed is inverse proportional. So, the 512x512 region size has been chosen for the feature extraction as a compromise between speed and accuracy.

V. Conclusion and Discussion

In this paper, we propose a novel technique to face the problem of the discrimination of CG and photographic images. We tried to capture the differences that exist in YCbCr space between the two classes by predicting their DC and AC coefficients, after DCT, by using their local neighbors, in different compression scales. In this way, it becomes possible to identify their different statistical characteristics. The feature vector that was selected, provides us with an accuracy of 93.14%, a result that indicates the reliability of the proposed method. In our future work, we plan to further explore the relationship that exists between the neighboring blocks in DCT.

VI. Acknowledgments

We would like to thank H. Farid and E. Tokuda et al., for sharing with us their databases of images and contributing in this way to improve our model.

Figure 5: Example PH image

Figure 6: Example CG image

References

[1] T. Whitted, "An improved illumination model for shaded display," *Commun. ACM*, vol. 23, pp. 343–349, June 1980.
[2] "Maya," http://usa.autodesk.com/maya/.
[3] "3ds max," http://usa.autodesk.com/3ds-max/.
[4] V. Schetinger, M. M. de Oliveira Neto, R. da Silva, and T. J. Carvalho, "Humans are easily fooled by digital images," *Computers & Graphics*, vol. 68, pp. 142–151, 2017.
[5] A. Rocha, W. J. Scheirer, T. Boult, and S. Goldenstein, "Vision of the unseen: Current trends and challenges in digital image and video forensics," *ACM Comput. Surv.*, vol. 43, p. 26, Oct 2011.
[6] T.-T. Ng, S. Chang, C.-Y. Lin, and Q. Sun, "Passive-blind image forensics," *Multimedia Security Technologies for Digital Rights Management*, July 2006.
[7] K. Rao and P. Yip., *Discrete Cosine Transform, Algorithms, Advantages, Applications*. Academic Press, 1990.
[8] E. Tokuda, H. Pedrini, and A. Rocha, "Computer generated images vs. digital photographs: A synergetic feature and classifier combination approach," *Journal of Visual Communication and Image Representation*, vol. 24, p. 1276–1292, Nov 2013.
[9] S. Lyu and H. Farid, "How realistic is photorealistic?" *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, Feb 2005.
[10] S. Fan, R. Wang, Y. Zhang, and K. Guo, "Classifying computer generated graphics and natural images based on image contour information," *Journal of Information and Computational Science*, vol. 9, pp. 2877–2895, Oct 2012.
[11] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui, "Physics-motivated features for distinguishing photographic images and computer graphics," in *Proceedings of the 13th Annual ACM International Conference on Multimedia*, ser. MULTIMEDIA '05. New York, NY, USA: ACM, 2005, pp. 239–248.
[12] Y. Tsin, V. Ramesh, and T. Kanade, "Statistical calibration of ccd imaging process," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 1, July 2001, pp. 480–487 vol.1.
[13] F. Peng and D.-l. Zhou, "Discriminating natural images and computer generated graphics based on the impact of cfa interpolation on the correlation of prnu," *Digital Investigation*, vol. 11, June 2014.
[14] A. C. Gallagher, "Image authentication by detecting traces of demosaicing," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2008, pp. 1–8.
[15] G. K. Wallace, "The jpeg still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii–xxxiv, Feb 1992.
[16] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," 2003.
[17] T. Fawcett, "Introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, June 2006.