

Dataset for classification of computer graphic images and photographic images

Halaguru Basavarajappa Basanth Kumar¹, Haranahalli Rajanna Chennamma²

¹DoS in Computer Science, SBRR Mahajana First Grade College (Autonomous), PG Wing, Mysuru, India

²Department of Computer Applications, JSS Science and Technology University, Mysuru, India

Article Info

Article history:

Received May 16, 2021

Revised Dec 24, 2021

Accepted Dec 30, 2021

Keywords:

Computer graphic images

Conventional machine learning

Deep learning

Handcrafted features

Photographic images

Photo-realistic computer graphics

ABSTRACT

The recent advancements in computer graphics (CG) image rendering techniques have made it easy for the content creators to produce high quality computer graphics similar to photographic images (PG) confounding the most naïve users. Such images used with negative intent, cause serious problems to the society. In such cases, proving the authenticity of an image is a big challenge in digital image forensics due to high photo-realism of CG images. Existing datasets used to assess the performance of classification models are lacking with: (i) larger dataset size, (ii) diversified image contents, and (iii) images generated with the recent digital image rendering techniques. To fill this gap, we created two new datasets, namely, 'JSSSTU CG and PG image dataset' and 'JSSSTU PRCG image dataset'. Further, the complexity of the new datasets and benchmark datasets are evaluated using handcrafted texture feature descriptors such as gray level co-occurrence matrix, local binary pattern and VGG variants (VGG16 and VGG19) which are pre-trained convolutional neural network (CNN) models. Experimental results showed that the CNN-based pre-trained techniques outperformed the conventional support vector machine (SVM)-based classifier in terms of classification accuracy. Proposed datasets have attained a low f-score when compared to existing datasets indicating they are very challenging.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Haranahalli Rajanna Chennamma

Department of Computer Applications, JSS Science and Technology University

Mysuru, India

Email: hrchennamma@jssstuniv.in

1. INTRODUCTION

Rapid advancement in computer graphic (CG) image rendering techniques give birth to applications such as animations, cartoons, gaming, photo-realism, virtual reality and many more [1]. Sophisticated CG software tools allow the user to produce synthetic images which are close to the reality and are difficult to identify whether an image is a camera captured or computer generated [2], [3]. If such images are used illegally in the court-of-law, journalism, criminal investigation, and political propaganda, then it may cause serious threat to the society [4]. In such cases, verifying the authenticity of images is a big challenge in digital image forensics.

To classify photo-realistic computer graphics (PRCG) from photographic images (PG), three benchmark datasets as shown in Table 1 are used in the literature: i) Columbia Photographic Images and Photorealistic Computer Graphics Dataset–Columbia Dataset; ii) DSTok dataset; and iii) A dataset created by Rahmouni *et al.*, to evaluate the performance of classification models. The DSTok dataset is the largest dataset in the literature with 9,700 samples. The aforementioned datasets are lacking with diversified image contents, sample size and more importantly the images in these datasets were produced/captured using older

versions of CG software's/camera models. Advancement in digital image rendering techniques have made it easy for the users to capture high quality images with regard to photographs and to produce photograph-like images which cannot be compared with the graphical contents produced using older versions of CG image rendering techniques. Hence, it is needed to upgrade datasets as well to evaluate updated innovations in the field of CG and PG image classification.

Table 1. Comparison of existing CG and PG image datasets.

Dataset	Year	CG Images	PG Images	Dataset Size	Publicly Available?	Limitations
Columbia Dataset [5]	2004	800	Personal: 800 Google: 800 Recaptured PRCG: 800	3,200	Yes	Small dataset and images are restricted to few categories.
DSTok Dataset [6]	2013	4,850	4,850	9,700	No	PRCG images in CG image class are relatively small.
Rahmouni <i>et al.</i> [7]	2017	1,800	1,800	3,600	Yes	Small dataset and CG image class consist of only video game screenshots.

In this paper, we propose two new datasets, namely “JSSSTU CG and PG image dataset” and “JSSSTU PRCG image dataset”. Initially the dataset is created with the intention of having diversified image contents with respect to CG and PG image categories and comprises 14,000 samples. Later dataset is created with the intention of having only photo-realistic computer graphics which are hard to distinguish with naked eyes that consist of 2,000 samples. Our new datasets would become very challenging and will be helpful for the researchers to develop efficient or improved classification models to produce better results who are working on the cutting-edge research problem: “classification of computer graphic images and photographic images”. Researchers have addressed this problem in different perspectives based on conventional machine learning and deep learning approaches.

a. Conventional machine learning

Significant improvement has been made in recent years to classify CG and PG images. Existing conventional machine learning techniques can be grouped into three categories based on the features selected for classification. They are: i) camera- characteristic based approaches [4], [8]–[12]; ii) spatial feature based approaches [13]–[20], and iii) geometric feature based approaches [21]–[28].

– Camera-characteristic based approaches

Techniques used to generate CG and PG images, undergo different pipeline architectures. Since PG images are acquired using digital cameras, they must exhibit distinct intrinsic properties which are not present in CG images. Based on this fact, some identification approaches have been described in [8]–[10]. Dehnie *et al.* [8] employed pattern noise caused due to the defect in camera sensors for classification of CG and PG images. Dirik *et al.* [9] proposed the features to detect the traces of color filter array (CFA) and chromatic aberration to distinguish CG and PG images. Khanna *et al.* [10] described a method based on residual pattern noise to distinguish scanner, CG and PG images. Photo response non uniformity (PRNU) noise is used as a digital fingerprint to identify the source camera in digital forensics and this is exploited in [4], [11], [12]. Peng *et al.* [11] proposed a method based on the theory of multifractal spectrum and features of PRNU, multifractal spectrum features of PRNU are extracted from an image to distinguish PRCG and PG images. Peng and Zhou [4] examine the changes in PRNU correlations and histogram features extracted from variance histograms of PRNU are used for identification of CG and PG images. Long *et al.* [12] proposed a method based on binary measures computed from PRNU noise in RGB channels to depict the differences between CG and PG images.

– Spatial feature based approaches

Pan *et al.* [13] show that the perceptual difference between CG and PG images is generally present in color and coarseness. Former is represented using fractal dimensions and the latter is described using generalised dimensions. Wu *et al.* [14] compute the difference in histogram of images and some higher histogram bins are considered as features to perform classification. Local binary pattern (LBP) [15] is a texture descriptor majorly used in image texture analysis and this is employed in [16], [17] to classify CG and PG images. Peng *et al.* [18] proposed a method based on statistical and textural features. Tan *et al.* [19] presented a novel scheme using local ternary count (LTC) which produces 54 dimensions of features from normalized histograms. Peng *et al.* [20] proposed a hybrid feature by analysing the differences in textures of residuals of CG and PG images.

– Geometric feature based approaches

Wang and Moulin [21] used wavelet-coefficients histogram as features extracted from wavelet-based statistical model for discrimination of CG and PG images. Chen *et al.* [22] built an alpha-stable model to describe wavelet decomposition coefficients of PG images. Wavelet domain is used to extract fractional lower order moments in images. Zhang and Wang [23] found that imaging features and visual features for images produced using different image acquisition processes reveal different statistical regularities in the wavelet domain. Based on this principle, statistical features and cross correlation of wavelet coefficients are used as features extracted from each sub-bands. Guo and Wang [24] presented a method based on multiwavelets which extracts the features in wavelet subbands. Fan *et al.* [25] proposed a scheme based on modified image contour transform in HSV color space to classify CG and PG images. Statistics such as average value, variance, skewness and kurtosis are computed in the wavelet domain. Birajdar and Mankar [26] used discrete wavelet transform to extract binary statistical image features by decomposing an image into subbands. Then the fuzzy entropy measure is employed to select relevant features. Quaternion wavelet transform is presented in [27] and [28], which extracts statistical features to classify CG and PG.

b. Deep learning approaches

Rahmouni *et al.* [7] proposed a novel scheme which combines statistical feature extraction to a convolutional neural network (CNN) architecture, then class label of the entire image is predicted by using weighted voting scheme which aggregates the local estimates of the class probabilities. Nguyen *et al.* [29] customized VGG-19 architecture to extract the generic features in the first three convolutional layers, and then statistical pooling layer is constructed as proposed in [7]. Pre-trained CNN models are employed in [30]–[33], and fine-tuned through transfer learning for binary classification. Chawla *et al.* [34] proposed five layers CNN architecture by introducing a special layer which takes some prediction error filters onto the first convolutional layer to ensure the correlation between pixels in PG and CG images. To predict the outcome of the original picture, two methods are used namely, weighted voting scheme and majority voting scheme. Former is used to label the image by aggregating the class probabilities and latter is used where the label is considered that appear in the majority of the image patches. Yao *et al.* [35] employed three sorts of high-pass filters to extract sensor noise residuals then piped into the proposed five layers CNN framework. Quan *et al.* [36] proposed a new CNN framework with two CNN cascaded convolutional layers at the end of the network. He *et al.* [37] described a novel deep learning approach by combining CNN and recurrent neural network (RNN). Thereafter, He *et al.* [38] proposed an attention-based dual-branch CNN to extract the features from combined color components. Meena and Tyagi [39] proposed an ensemble model by combining the features produced by VGG-19 pre-trained CNN and noise features produced using high-pass filters to discriminate CG and PG images.

From the above study, even though much progress has been made for the classification of CG and PG images, existing techniques and datasets used to evaluate the performance is still have the following limitations: i) in the existing datasets, sample size and image contents are limited and do not make compelling high quality image content due to advancement in image rendering techniques; (ii) in prior works, accuracy of the classification model depends on choice of the feature descriptor used for classification.

In the task of distinguishing CG and PG images, the contributions of the paper are outlined as:

- Due to non-availability of large, heterogeneous dataset containing CG and PG images, ‘JSSSTU CG and PG image dataset’ is created. ‘JSSSTU PRCG image dataset’ is created which exhibits high photo-realism.
- Effectiveness of the existing texture based feature descriptors and CNN based deep learning techniques are investigated on our new datasets and benchmark datasets.

Remainder of this paper is organised as follows: section 2 presents a description of the state-of-the-art techniques based on conventional machine learning and deep learning. Section 3 demonstrates performance of the techniques on our new datasets and benchmark datasets through experimental results. Conclusion is given in section 4.

2. RESEARCH METHOD

In this section, state-of-the-art conventional machine learning and CNN based deep learning techniques are developed for the task of classifying CG and PG images. Handcrafted textural features are considered for conventional machine learning and VGG variants CNN based pre-trained models, are used for deep learning. They are described in the following sections.

2.1. Conventional machine learning techniques

Texture describes surface characteristics of an image. Most widely used texture descriptors such as gray level co-occurrence matrix (GLCM) [40] and LBP [15] are employed to analyse the surface texture of CG and PG image. Texture surface of CG image appear smoother than those of PG image, which exhibits the basic differences between them. Hence, the aforementioned texture descriptors are used in our work.

2.1.1. GLCM descriptor

GLCM is a statistical method which computes the occurrence of pairs of pixels or gray levels in a particular orientation over all in an image or image region. This is represented by using parameters (Θ , d) where ' Θ ' represents orientation and ' d ' is the distance between two picture elements. The GLCM descriptor allows rotational invariance and it is defined by 8 orientations separated by $\Pi/4$ radians. Haralick *et al.* [40] defined 14 statistical properties computed from the normalized GLCM matrix. In this work, we employed four properties such as contrast, correlation, energy and homogeneity.

- Contrast: contrast computes local intensity variation between a picture element and to its neighbor for the entire image as given in (2). The range is calculated using the (1).

$$\text{Range} = [0 \text{ (size(GLCM,1)-1)}^2] \quad (1)$$

Where, GLCM represents normalized matrix. I_{mn} and variables m and n from (2)-(5) represent $(m, n)^{\text{th}}$ entry and a value at (m, n) in a normalized GLCM.

$$\text{Contrast} = \sum_{m,n} |m - n|^2 I_{mn} \quad (2)$$

- Correlation: correlation computes correlation of a picture element to its neighbor over the entire image. It returns a value between -1 and 1 for a positively or negatively correlated image. Otherwise, return unrepresentative value for a constant image as given in (3). Where, ' μ ' and ' σ ' indicates mean and standard deviation of the marginal distributions associated with I_{mn}/R , and R is a normalized constant.

$$\text{Correlation} = \sum_{m,n} I_{mn} \frac{(m - \mu)(n - \mu)}{\sigma^2} \quad (3)$$

- Energy: energy is also termed as angular second moment which computes the sum of squared elements. It returns a value between 0 and 1, otherwise, returns 1 for a constant image as given in (4).

$$\text{Energy} = \sum_{m,n} (I_{mn})^2 \quad (4)$$

- Homogeneity: Homogeneity computes the closeness of elements diagonally in GLCM. It returns 1 for diagonal elements, otherwise, returns a value between 0 and 1 as given in (5).

$$\text{Homogeneity} = \sum_{m,n} \frac{I_{mn}}{1 + |m - n|} \quad (5)$$

2.1.2. LBP descriptor

LBP is a texture descriptor operator proposed by Ojala [15], which encodes each pixel value of an image by comparing its neighborhood pixels with the center pixel value. If the intensity of neighboring pixel is greater than or equal to the intensity of center pixel mark the neighboring pixel as 1, otherwise, mark as 0 which result in a binary sequence. Then, a bit vector is converted into decimal number and is replaced with center pixel value. LBP descriptor of every pixel in an image is computed using (6) and $f(s)$ is given in. (7).

$$\text{LBP}(N, R) = \sum_{N=0}^{N-1} f(I_n - I_c) 2^N \quad (6)$$

$$f(s) = \begin{cases} 1, & \text{if } s \geq 0 \\ 0, & \text{Otherwise.} \end{cases} \quad (7)$$

Where, I_n and I_c indicate intensity of neighboring and current pixel respectively. N represents the number of neighbors chosen at a radius of R . In this work, we choose $P=8$ neighbors with a radius $R=1$.

2.2. Support vector machines (SVM) classifier

SVM [41] are the most widely used and effective supervised machine learning algorithm for classification problems. It can be used to perform linear and non-linear classification. In this work, we perform non-linear classification, when the feature vectors could not be separated linearly. Radial basis function (RBF) kernel is chosen in the experimentation and it is described in (8). Where, $\|u_1 - u_2\|$ in (8) represent euclidean distance between two feature vector points u_1 and u_2 and ' σ ' represent variance.

$$k(u_1, u_2) = \exp\left(-\frac{\|u_1 - u_2\|^2}{2\sigma^2}\right) \quad (8)$$

2.3. CNN based deep learning techniques

Among other deep neural networks, CNN based deep learning techniques have shown its effectiveness by obtaining general features to specific features automatically based on the image content. CNN based pre-trained neural network models such as AlexNet [42], VGG (VGG16 and VGG19) [43], GoogLeNet [44], and ResNet [45], have shown great performance in classifying images into 1000 object categories such as pencil, keyboard, mouse, etc., and have become standard models for classification tasks. These models are trained on millions of images and have learnt rich feature representations for a wide range of PG images in the ImageNet database.

Training a deep ConvNet model from scratch takes several days or weeks or even months on a large dataset. A pre-trained neural network model would be a better choice to solve similar kinds of problems for a smaller dataset. In this work, two variants of VGG pre-trained neural network models such as VGG16 and VGG19 are adopted. Because, these neural network models have fixed size kernels which take less time to process and easily capture small patterns. Transfer learning is applied to perform classification on a new dataset which contains CG and PG images. It can be carried out in two ways: feature extraction and fine-tuning. Latter is adopted in our work and is performed by replacing the last three layers of pre-trained neural network models and these layers are fine-tuned for classification of CG images and PG images.

2.3.1. Visual geometry group (VGG) architecture

VGG architecture can be viewed as an input layer, feature extraction layers and classification layers. In the input layer, a color image of fixed size 227×227 is input to the architecture during training. The image is pre-processed by subtracting the mean RGB value from each pixel on the training set. During feature extraction, the image is moved through a series of convolutional layers, where the filters of fixed size 3×3 are used. Spatial padding and stride is fixed to 1 pixel which preserves the spatial dimension after convolution. The depth of the convolutional layers begins from 64 in the first layer and increases by a factor of 2 after every maximum pooling layer until it attains 512. Spatial dimension of the image is reduced by maximum pooling layers and this is done by using a filter of size 2 and a stride of 2. Five maximum pooling layers are used in the architecture which follows some convolutional layers. Classification layers consist of three fully connected layers: first two have 1024 neurons each and third contain two neurons and at the end sigmoid activation function is used to perform binary classification which produces the value in the range 0 and 1 and it is described in (9). VGG variants CNN architecture is presented in Table 2 (the parameters of convolutional layers are denoted as conv(block)-(number of filters)_ layer number at each block. ReLU is not shown for brevity).

$$s(t) = \frac{1}{1 + e^{-x}} \quad (9)$$

For training, binary cross entropy loss function is used and is given in (10).

$$\text{Binary cross entropy} = -\frac{1}{M} \sum_{j=1}^M -(y_j \times \log(p_j) + (1 - y_j) \times \log(1 - p_j)) \quad (10)$$

Where, M is the number of categories, $(1 - p_j)$ is the probability of class CG. p_j is the probability of class PG and y is the binary indicator (0 or 1) if category label is the correct classification for sample. Rectified linear unit (ReLU), a non-linear activation function, i.e. $f(x) = \max(0, x)$ is used in all hidden layers of VGG variants.

Table 2. VGG variant CNN configuration: output volume and parameters for VGG16 and VGG19 architecture

VGG variant CNN configuration					
VGG16	Output volume	Parameters	VGG19	Output volume	Parameters
Image input (227×227 color image)					
conv1-64_1	227×227×64	1792	conv1-64_1	227×227×64	1792
conv1-64_2	227×227×64	36928	conv1-64_2	227×227×64	36928
Maximum pooling					
conv2-128_1	113×113×128	73856	conv2-128_1	113×113×128	73856
conv2-128_2	113×113×128	147584	conv2-128_2	113×113×128	147584
Maximum pooling					
conv3-256_1	56×56×256	295168	conv3-256_1	56×56×256	295168
conv3-256_2	56×56×256	590080	conv3-256_2	56×56×256	590080
conv3-256_3	56×56×256	590080	conv3-256_3	56×56×256	590080
			conv3-256_4	56×56×256	590080
Maximum pooling					
conv4-512_1	28×28×512	1180160	conv4-512_1	28×28×512	1180160
conv4-512_2	28×28×512	2359808	conv4-512_2	28×28×512	2359808
conv4-512_3	28×28×512	2359808	conv4-512_3	28×28×512	2359808
			conv4-512_4	28×28×512	2359808
Maximum pooling					
conv5-512_1	14×14×512	2359808	conv5-512_1	14×14×512	2359808
conv5-512_2	14×14×512	2359808	conv5-512_2	14×14×512	2359808
conv5-512_3	14×14×512	2359808	conv5-512_3	14×14×512	2359808
			conv5-512_4	14×14×512	2359808
Maximum pooling					
FC – 1024					
FC – 1024					
FC – 1					
Sigmoid					
Trainable parameters (in millions)	-	41,343,873	-	-	46,653,569

3. RESULTS AND DISCUSSION

3.1. Dataset collection

‘JSSSTU CG and PG image dataset’ consists of image categories: CG and PG images with 7,000 samples in each class, containing diversified contents. CG images are collected from various reliable computer graphics websites and PG images are captured from different camera models (standalone, in-built mobile cameras) as the camera specifications for each model vary in terms of megapixel count, image quality, sensor type and so on. To improve the diversity of PG image contents, they are collected from other sources INRIA [46], ICCV09 [47], and McGill calibrated colour image database [48]. Contents of the CG image class include 3D model, architecture, cartoon, digital art, non-PRCG images, object, people, PRCG images, texture, trademark, vector maps and video gaming. PG image class cover a wide range of image contents: animals, buildings, man-made objects, indoor scenes, outdoor scenes, nature, vehicles and so on. ‘JSSSTU PRCG image dataset’ contains 2,000 samples which demonstrate high photo-realism.

Online sources used to create CG image dataset are given in Table 3. Camera models and other sources used to create PG image dataset are given in Tables 4 and 5. Various online sources used to create PRCG image dataset are given in Table 6. The aforementioned datasets are made publicly available to the research community at the following link: <https://sites.google.com/view/hrchennamma/research-activities/jssstu-data-sets>.

Table 3. Online sources used to create CG image dataset

Image class	Online sources		
Computer graphic images	https://pixbay.com	https://www.grabcad.com	http://www.3dlinks.com
	http://wallpaperlepi.com	https://freestocktextures.com	http://www.realsoft.com
	http://www.cgw.com	http://www.acitymap.com	www.digitalrepose.com
	http://www.cgsociety.org	http://www.cadnav.com	www.google.com/imghp?hl=EN
	https://free3d.com	https://www.nexusmods.com	
	http://fantasyartdesign.com	https://wallpaper.mob.org	

In addition to these datasets, existing benchmark datasets presented in [5]–[7], are used for the experimentation. Sample size pertaining to each class and each dataset are shown in Table 7. Image samples

from JSSSTU CG and PG image dataset and JSSSTU PRCG image dataset is shown in Figures 1(a)-(c) respectively.

All the images are resized to a dimension of 227×227 pixels. Textual information present in some of the computer graphic images is cropped. Datasets are randomly partitioned into 80% for training (70% for training and 10% for validation in case of pre-trained neural network model) and 20% for testing.

Table 4. Camera models used to create PG image dataset

Image class	Camera models
Photographic images	Canon PowerShot A2200, NIKON D7100, NokiaC6-01, Canon PowerShot SX200 IS, Vivo 1714, Canon PowerShot SD1100 IS, SAMSUNG GT-S7262, Lenovo K50a40, SONY DSC-WX7, Vivo 1718, Canon PowerShot A400, Canon EOS 1100D, Canon EOS 1000D.

Table 5. PG image sources used to create PG image dataset

PG image source	Count
Personal collection	4,019
INRIA	1,280
ICCV09	622
McGill	1,079
Total	7,000

Table 6. Online sources used to create JSSSTU PRCG image dataset

Image class	Online sources		
PRCG images	https://www.3dartistonline.com	https://www.gamespot.com	https://www.chaosgroup.com
	https://3dexport.com	https://gizmodo.com	https://app.easyrender.com
	https://archicgi.com	http://www.graphicmania.net	https://evermotion.org
	https://archvizcamp.com	https://lumion.com	https://www.freepik.com
	https://area.autodesk.com	https://www.maxon.net	https://www.ronenbekerman.com
	https://www.blenderguru.com	http://www.nextlimit.com	https://www.thearender.com
	https://www.cgmeetup.com	https://www.blog.poliigon.com	https://forums.unrealengine.com
	https://www.cgtrader.com	https://www.vizpark.com	http://www.3dlinks.com

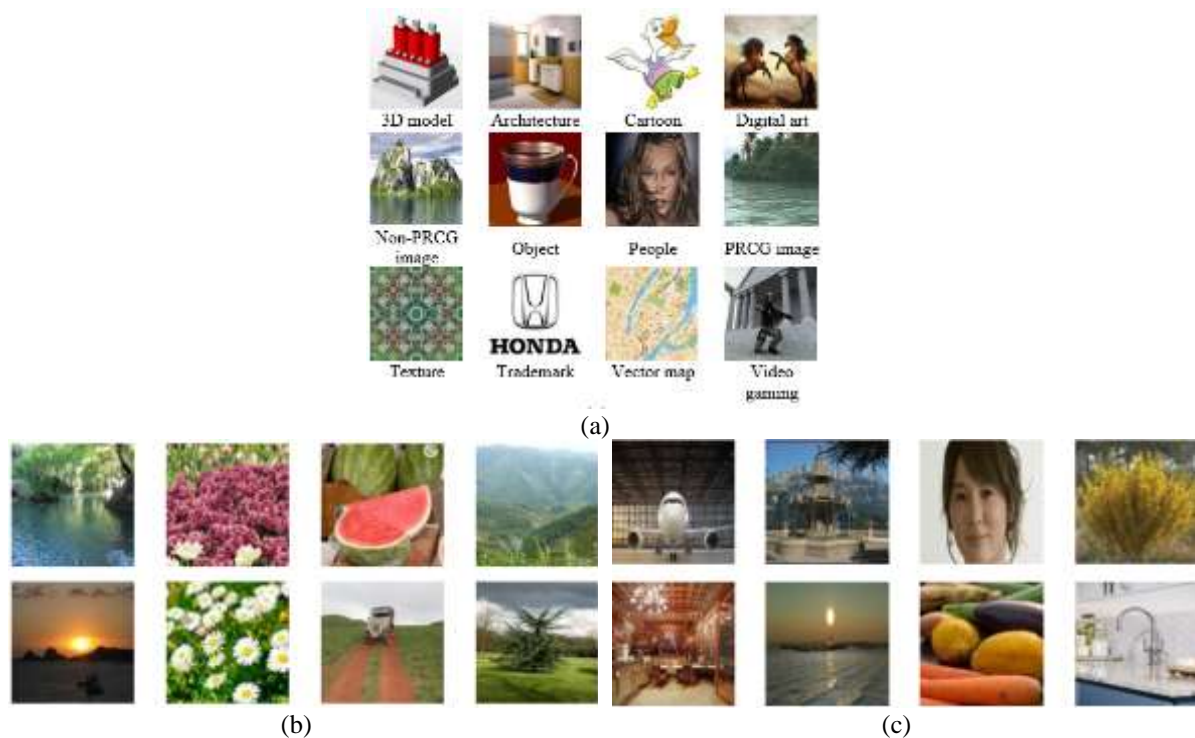


Figure 1. Image samples from JSSSTU CG and PG image dataset and JSSSTU PRCG image dataset (a) CG images, (b) PG images, and (c) PRCG images

Table 7. Datasets used in the experiment

Dataset	CG images	PG images	Dataset size
Columbia dataset	800	800	1,600
DSTok dataset	4,850	4,850	9,700
Rahmouni <i>et al.</i> dataset	1,800	1,800	3,600
JSSSTU CG and PG image dataset proposed	7,000	7,000	14,000
JSSSTU PRCG image dataset proposed	2,000	2,000	4,000

3.2. Experiments

3.2.1. Experiment setup for conventional machine learning techniques

Experiments are conducted using MATLAB R2018a with Intel Core-i3 4005U processor, 1.70 GHz and 8 GB RAM. Texture features such as GLCM and LBP are extracted independently from an image which consists of a feature dimension 4 and 59 respectively. SVM classifier is used in the experimentation.

3.2.2. Experiment setup for pre-trained neural network models

VGG variants (VGG16 and VGG19) are implemented with Google Colaboratory platform on the free ‘Tesla K80 GPU’ with 25 GB RAM using Keras. Regularization techniques such as early stopping (monitor=validation loss and patience=10), data augmentation and dropout (dropout probability 0.5) [49] are used to prevent the pre-trained neural network models from overfitting. Hyper-parameters such as stochastic gradient descent (SGD) optimizer with default momentum value of 0.9 and maximum number of epochs 100 are used for all datasets during training. Other hyper-parameters such as batch size and initial learn rate and learn rate drop factor used for different datasets are given:

- Columbia dataset, DSTok Dataset, Rahmouni *et al.* Dataset and JSSSTU CG and PG image dataset
VGG variants (VGG16 and VGG19): batch size of 32 images, an initial learn rate of 1e-4, learn rate drop factor (monitor=validation loss, factor=0.1, and patience=5) are used.
- JSSSTU PRCG image dataset
VGG variants (VGG16 and VGG19): batch size of 16 images, an initial learn rate of 1e-4 and learn rate drop factor (monitor=validation loss, factor=0.1, and patience=5) are used.
- Data augmentation
To increase the diversity in content of images, we employed data augmentation techniques such as translation, rotation, shear, reflection and zooming. These techniques are used during training of VGG16 and VGG19 pre-trained CNN on JSSSTU CG and PG image dataset, JSSSTU PRCG image dataset and DSTok Dataset respectively. Aforementioned random transformations help the model to expose to more aspects of data and yield better generalization.

3.2.3. Experiment results

Average classification accuracies obtained using handcrafted texture features and pre-trained CNN on our new datasets are tabulated in Table 8. As shown in Table 8, CNN based pre-trained techniques outperformed the classification accuracy performance against the conventional SVM-based classifier. VGG19 has attained better classification results when compared to the handcrafted texture features and VGG16.

Table 8. Average classification accuracies of handcrafted texture features and pre-trained CNN on our new datasets

Dataset	Average classification accuracy in %			
	Handcrafted features		Pre-trained CNN	
	GLCM	LBP	VGG16	VGG19
JSSSTU CG and PG image dataset	63.67	85.75	94.42	94.46
JSSSTU PRCG image dataset	63.50	75.62	89.12	89.37

3.2.4. Comparative analysis of benchmark datasets used to evaluate classification models

Existing methods based on conventional machine learning and deep learning are used to compare their performances on existing benchmark datasets. Average accuracies attained are listed in Table 9. Performance of the methods is given from highest to lowest. As seen from Table 9, VGG19 pre-trained CNN has achieved cent percent classification results on Columbia dataset. Feature fusion method based on conventional machine learning proposed by Tokuda *et al.* has obtained better identification accuracy on DSTok dataset. Further, techniques presented in [29], [34], [35] have attained cent percent accuracy on a Rahmouni *et al.* dataset.

Table 9. Comparative analysis of benchmark datasets used to evaluate classification models

Columbia dataset		DSTok dataset		Rahmouni <i>et al.</i> dataset	
Method	Average accuracy (%)	Method	Average accuracy (%)	Method	Average accuracy (%)
VGG19	100	Tokudal <i>et al.</i> [6]	97	Yao <i>et al.</i> [35]	100
VGG16	99.37	Ming He [30]	96	Chawla <i>et al.</i> [34]	100
Cui <i>et al.</i> [31]	98	Rezende <i>et al.</i> [33]	94	Nguyen <i>et al.</i> [29]	100
LBP	97.50	VGG19	90.92	VGG19	95.96
Fan <i>et al.</i> [25]	93.51	VGG16	89.94	VGG16	95.42
GLCM	78.75	LBP	75.61	Rahmouni <i>et al.</i> [7]	93.2
		GLCM	61.85	LBP	87.22
				GLCM	66.52

3.2.5. Performance metrics

The metrics such as precision, recall and f-score [50] are used to assess the performance of VGG19 pre-trained neural network model as it yields best classification accuracy against handcrafted features and VGG16 pre-trained neural network model on existing and proposed datasets. Macro average of aforementioned metrics is computed for two classes. Table 10 shows the evaluation metrics used to assess the performance of VGG19 pre-trained neural network model on existing and proposed datasets.

As seen from Table 10, low f-score is obtained on JSSSTU PRCG image dataset when compared to other datasets. The difference in f-score of JSSSTU CG and PG image dataset and DSTok dataset is only 0.3. Hence, we conclude that, our new datasets are very challenging and the DSTok dataset is as good as JSSSTU CG and PG image dataset but it is lacking with larger dataset size, contain limited number of PRCG images and images produced using recent rendering technology.

Table 10. Performance metrics used to assess the performance of VGG19 pre-trained neural network model on different datasets

Dataset	Performance metrics		
	Precision	Recall	F-score
Columbia Dataset	1	1	1
Rahmouni <i>et al.</i> Dataset	0.96	0.96	0.96
JSSSTU CG and PG image dataset – Proposed	0.94	0.94	0.94
DSTok Dataset	0.91	0.91	0.91
JSSSTU PRCG image dataset – Proposed	0.89	0.89	0.89

4. CONCLUSION

This work is aimed at creating two new datasets, namely ‘JSSSTU CG and PG image dataset’ a heterogeneous dataset which comprises 14,000 samples and ‘JSSSTU PRCG image dataset’ which exhibits photo-realism with 2,000 samples. Further, we implemented state-of-the-art techniques based on handcrafted texture features and deep learning. Performance of these techniques is evaluated on our new datasets and benchmark datasets. Experimental results show that CNN based pre-trained techniques outperformed the classification accuracy performance against the conventional SVM-based classifier. Further, we found that the choice of handcrafted features used for classification has achieved better results on the Columbia Dataset when compared to other benchmark datasets and our new datasets. The performance of VGG19 pre-trained neural network technique has attained significant results on ‘JSSSTU CG and PG image dataset’ but still the accuracy can be improved. On the other hand, its performance on ‘JSSSTU PRCG image dataset’ has achieved low detection rate due to the high-realism images present in the dataset. Hence, an efficient and robust technique is needed to solve this problem and our new datasets will be helpful for the researchers who are working on the cutting-edge research problem: “classification of computer graphic images and photographic images” to evaluate their classification models. To the best of our knowledge, these kinds of datasets do not exist in the literature.

REFERENCES





- [1] “Applications of computer graphics.” <https://www.geekforgeeks.org/applications-of-computer-graphics/> (accessed Jan. 05, 2021).
- [2] O. Holmes, M. S. Banks, and H. Farid, “Assessing and improving the identification of computer-generated portraits,” *ACM Transactions on Applied Perception*, vol. 13, no. 2, pp. 1–12, Mar. 2016, doi: 10.1145/2871714.
- [3] S. Lyu and H. Farid, “How realistic is photorealistic?,” *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, Feb. 2005, doi: 10.1109/TSP.2004.839896.
- [4] F. Peng and D. Zhou, “Discriminating natural images and computer generated graphics based on the impact of CFA interpolation on the correlation of PRNU,” *Digital Investigation*, vol. 11, no. 2, pp. 111–119, Jun. 2014, doi: 10.1016/j.diin.2014.04.002.
- [5] T. Ng, S. Chang, J. Hsu, and M. Pepeljugoski, “Columbia photographic images and photorealistic computer graphics dataset,” *Columbia University, ADVENT Technical Report*, pp. 1–23, 2004.

- [6] 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, no. 8, pp. 1276–1292, Nov. 2013, doi: 10.1016/j.jvcir.2013.08.009.
- [7] N. Rahmouni, V. Nozick, J. Yamagishi, and I. Echizen, "Distinguishing computer graphics from natural images using convolution neural networks," in *2017 IEEE Workshop on Information Forensics and Security, WIFS 2017*, Dec. 2017, pp. 1–6, doi: 10.1109/WIFS.2017.8267647.
- [8] S. Dehnie, T. Sencar, and N. Memon, "Digital image forensics for identifying computer generated and digital camera images," in *2006 International Conference on Image Processing*, Oct. 2006, pp. 2313–2316, doi: 10.1109/ICIP.2006.312849.
- [9] A. E. Dirik, S. Bayram, H. T. Sencar, and N. Memon, "New features to identify computer generated images," in *2007 IEEE International Conference on Image Processing*, 2007, vol. 4, pp. 433–436, doi: 10.1109/ICIP.2007.4380047.
- [10] N. Khanna, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Mar. 2008, pp. 1653–1656, doi: 10.1109/ICASSP.2008.4517944.
- [11] F. Peng, J. Shi, and M. Long, "Identifying photographic images and photorealistic computer graphics using multifractal spectrum features of PRNU," in *2014 IEEE International Conference on Multimedia and Expo (ICME)*, Jul. 2014, pp. 1–6, doi: 10.1109/ICME.2014.6890296.
- [12] M. Long, F. Peng, and Y. Zhu, "Identifying natural images and computer generated graphics based on binary similarity measures of PRNU," *Multimedia Tools and Applications*, vol. 78, no. 1, pp. 489–506, Jan. 2019, doi: 10.1007/s11042-017-5101-3.
- [13] F. Pan, J. Chen, and J. Huang, "Discriminating between photorealistic computer graphics and natural images using fractal geometry," *Science in China, Series F: Information Sciences*, vol. 52, no. 2, pp. 329–337, Feb. 2009, doi: 10.1007/s11432-009-0053-5.
- [14] R. Wu, X. Li, and B. Yang, "Identifying computer generated graphics VIA histogram features," in *2011 18th IEEE International Conference on Image Processing*, Sep. 2011, pp. 1933–1936, doi: 10.1109/ICIP.2011.6115849.
- [15] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.
- [16] Z. Li, J. Ye, and Y. Q. Shi, "Distinguishing computer graphics from photographic images using local binary patterns," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7809, Springer Berlin Heidelberg, 2013, pp. 228–241.
- [17] Z. Li, Z. Zhang, and Y. Shi, "Distinguishing computer graphics from photographic images using a multiresolution approach based on local binary patterns," *Security and Communication Networks*, vol. 7, no. 11, pp. 2153–2159, Nov. 2014, doi: 10.1002/sec.929.
- [18] F. Peng, J. ting Li, and M. Long, "Identification of natural images and computer-generated graphics based on statistical and textural features," *Journal of Forensic Sciences*, vol. 60, no. 2, pp. 435–443, Mar. 2015, doi: 10.1111/1556-4029.12680.
- [19] D. Q. Tan, X. J. Shen, J. Qin, and H. P. Chen, "Detecting computer generated images based on local ternary count," *Pattern Recognition and Image Analysis*, vol. 26, no. 4, pp. 720–725, Oct. 2016, doi: 10.1134/s1054661816040167.
- [20] F. Peng, D. Zhou, M. Long, and X. Sun, "Discrimination of natural images and computer generated graphics based on multifractal and regression analysis," *AEU - International Journal of Electronics and Communications*, vol. 71, pp. 72–81, Jan. 2017, doi: 10.1016/j.aeu.2016.11.009.
- [21] Y. Wang and P. Moulin, "On discrimination between photorealistic and photographic images," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2006, vol. 2, pp. II-161–II-164, doi: 10.1109/icassp.2006.1660304.
- [22] D. Chen, J. Li, S. Wang, and S. Li, "Identifying computer generated and digital camera images using fractional lower order moments," in *2009 4th IEEE Conference on Industrial Electronics and Applications, ICIEA 2009*, May 2009, pp. 230–235, doi: 10.1109/ICIEA.2009.5138202.
- [23] R. Zhang and R. Wang, "Distinguishing photorealistic computer graphics from natural images by imaging features and visual features," in *2011 International Conference on Electronics, Communications and Control (ICECC)*, Sep. 2011, pp. 226–229, doi: 10.1109/ICECC.2011.6067631.
- [24] K. Guo and R. Wang, "A new method for detecting computer-generated images based on multiwavelets," *Journal of Information and Computational Science*, vol. 8, no. 8, pp. 1449–1456, 2011.
- [25] 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, no. 10, pp. 2877–2895, 2012.
- [26] G. K. Birajdar and V. H. Mankar, "Computer graphic and photographic image classification using local image descriptors," *Defence Science Journal*, vol. 67, no. 6, pp. 654–663, Nov. 2017, doi: 10.14429/dsj.67.10079.
- [27] J. Wang, T. Li, Y.-Q. Shi, S. Lian, and J. Ye, "Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics," *Multimedia Tools and Applications*, vol. 76, no. 22, pp. 23721–23737, Nov. 2017, doi: 10.1007/s11042-016-4153-0.
- [28] J. Wang, T. Li, X. Luo, Y. Q. Shi, and S. K. Jha, "Identifying computer generated images based on quaternion central moments in color quaternion wavelet domain," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 9, pp. 2775–2785, Sep. 2019, doi: 10.1109/TCSVT.2018.2867786.
- [29] H. H. Nguyen, N. D. T. Tieu, H. Q. Nguyen-Son, V. Nozick, J. Yamagishi, and I. Echizen, "Modular convolutional neural network for discriminating between computer-generated images and photographic images," in *ACM International Conference Proceeding Series*, Aug. 2018, pp. 1–10, doi: 10.1145/3230833.3230863.
- [30] M. He, "Distinguish computer generated and digital images: a CNN solution," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 12, Jun. 2019, doi: 10.1002/cpe.4788.
- [31] Q. Cui, S. McIntosh, and H. Sun, "Identifying materials of photographic images and photorealistic computer generated graphics based on deep CNNs," *Computers, Materials and Continua*, vol. 55, no. 2, pp. 229–241, 2018, doi: 10.3970/cmc.2018.01693.
- [32] K. B. Meena and V. Tyagi, "A deep learning based method to discriminate between photorealistic computer generated images and photographic images," in *Communications in Computer and Information Science*, Springer Singapore, 2020, pp. 212–223.
- [33] E. R. S. de Rezende, G. C. S. Ruppert, A. Theóphilo, E. K. Tokuda, and T. Carvalho, "Exposing computer generated images by using deep convolutional neural networks," *Signal Processing: Image Communication*, vol. 66, pp. 113–126, Aug. 2018, doi: 10.1016/j.image.2018.04.006.
- [34] C. Chawla, D. Panwar, G. S. Anand, and M. P. S. Bhatia, "Classification of computer generated images from photographic images using convolutional neural networks," in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, Oct. 2018, pp. 1053–1057, doi: 10.1109/ICACCCN.2018.8748829.





- [35] Y. Yao, W. Hu, W. Zhang, T. Wu, and Y.-Q. Shi, "Distinguishing computer-generated graphics from natural images based on sensor pattern noise and deep learning," *Sensors*, vol. 18, no. 4, Apr. 2018, doi: 10.3390/s18041296.
- [36] W. Quan, K. Wang, D.-M. Yan, and X. Zhang, "Distinguishing between natural and computer-generated images using convolutional neural networks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2772–2787, Nov. 2018, doi: 10.1109/TIFS.2018.2834147.
- [37] P. He, X. Jiang, T. Sun, and H. Li, "Computer graphics identification combining convolutional and recurrent neural networks," *IEEE Signal Processing Letters*, vol. 25, no. 9, pp. 1369–1373, Sep. 2018, doi: 10.1109/LSP.2018.2855566.
- [38] P. He, H. Li, H. Wang, and R. Zhang, "Detection of computer graphics using attention-based dual-branch convolutional neural network from fused color components," *Sensors*, vol. 20, no. 17, pp. 1–15, Aug. 2020, doi: 10.3390/s20174743.
- [39] K. B. Meena and V. Tyagi, "Distinguishing computer-generated images from photographic images using two-stream convolutional neural network," *Applied Soft Computing*, vol. 100, Mar. 2021, doi: 10.1016/j.asoc.2020.107025.
- [40] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 6, pp. 610–621, Nov. 1973, doi: 10.1109/TSMC.1973.4309314.
- [41] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/bf00994018.
- [42] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.
- [43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, Sep. 2015, [Online]. Available: <http://arxiv.org/abs/1409.1556>.
- [44] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
- [45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [46] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5302, Springer Berlin Heidelberg, 2008, pp. 304–317.
- [47] S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions," in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 1–8, doi: 10.1109/ICCV.2009.5459211.
- [48] "McGill calibrated colour image database," *McGill Vision Research*. <http://tabby.vision.mcgill.ca/html/browsedownload.html> (accessed Jan. 10, 2021).
- [49] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [50] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.

BIOGRAPHIES OF AUTHORS



Halaguru Basavarajappa Basanth Kumar     is currently working as Assistant Professor in the Department of Studies in Computer Science, SBRR Mahajana First Grade College (Autonomous), PG Wing, Mysuru. He received MCA from Visvesvaraya Technological University, Karnataka, India in 2007. He is currently pursuing Ph.D. at Sri Jayachamarajendra College of Engineering, Visvesvaraya Technological University, Karnataka, India. His research interest is machine learning. He can be contacted at email: basanth.10@gmail.com.



Haranahalli Rajanna Chennamma     received Ph.D. in Computer Science from the University of Mysore in the area of Digital Image Forensics in 2011. Subsequently, she served as a Post Doctoral fellow in the Department of Computer Science and Engineering, University of North Texas, USA in 2012. Currently, she is an Associate Professor in the Department of Computer Applications, JSS Science and Technology University, Mysuru. Previously, she has been awarded a prestigious Fellowship under the scheme of Research and Development in Forensics from the Department of Home Affairs, Government of India in 2005. Chennamma served as a Senior Research Fellow (SRF) in Central Forensic Science Laboratory, Hyderabad. She served as a Project Trainee for a year at the National Aerospace Laboratory (NAL), Bangalore and she also served as a software engineer for a year in a multinational software company, Bangalore. She is the recipient of two "Best Scientific Paper Awards". Her research areas of interest are Image and Video forensics, pattern recognition and Computer Vision. She can be contacted at email: hrchennamma@jssstuniv.in.