This paper presents a study of cell-phone camera model linkage that matches digital images against potential makes / models of cell-phone camera sources using camera color interpolation features. The matching performance is examined and the dependency on the content of training image collection is evaluated via variance analysis. Training content dependency can be dealt with under the framework of component forensics, where cell-phone camera model linkage is seen as a combination of semi non-intrusive training and completely non-intrusive testing. Such a viewpoint suggests explicitly the goodness criterion of testing accuracy for training data selection. It also motivates other possible alternative training procedures based on different criteria, such as the training complexity, for which preliminary but promising experiment designs and results have been obtained.

Index Terms—image forensics, cell-phone camera, color interpolation, non-intrusive forensics, content dependency.

1. INTRODUCTION

The past decade has seen tremendous growth in the number of mobile imaging devices due to the rapid advancement of visual sensor technologies. Digital images have become ubiquitous and have been used in applications from consumer photography to national security. As the importance of digital images grows, concerns regarding the origin and authenticity of digital images have also been raised and are receiving increasing attention. One can readily ask a series of forensic questions about a given digital image; for example, what kind of acquisition device was used to generate this image? If the image was taken by a camera, what is the make and model of the camera? Has this image undergone any non-trivial post-processing or manipulation?

All these questions lie under the umbrella of digital image forensics, which is now a very active research area. Extensive efforts have been made and various techniques and tools have been developed. For instance, Fridrich et al. [1] developed the methodology of exploiting the Photo-Response Non-Uniformity (PRNU) to distinguish different camera units. Swaminathan et al. [2] showed how to use the color interpolation coefficients to identify different camera models. Also, they employed blind deconvolution to estimate the linear and shift-invariant (LSI) part of the overall post-processing step and the estimate will be matched against an identity system to determine if there is any non-trivial manipulation. [3] Ng et al. [4] proposed physics-motivated features to separate real photos and computer graphics. Popescu and Farid [5] estimated the inter-pixel correlation caused by interpolation for detecting rescaling operations. Toward a unifying understanding of digital image forensics, a framework of component forensics has been established [6] for the study of more generic scenarios.

We consider in this paper the cell-phone camera model linkage problem that matches digital images against potential models of cell-phone camera sources. This problem finds it applications in many forensic and homeland-security scenarios. For example, a forensic analyst during an anti-terrorism act may find a cell-phone camera in which some images have been stored whose authenticity are to be examined. Clearly, the first thing the analyst can check is whether or not the images are from the exact cell-phone camera that is found, and this can be well achieved using features such as the Photo-Response Non-Uniformity (PRNU) [1] that captures the camera-specific characteristics. Camera model linkage serves as a crucial forensic means when the images are not from the exact cell-phone camera, in which case the camera brand or model that has been used to generate the images will be valuable information about the images’ possible origin.

The contributions of this paper are as follows. We thoroughly examine the color interpolation coefficients suggested by [2] as the feature for cell-phone camera model matching, and find that the selection of training data will have a critical impact on the achievable matching accuracy. As training input data can be collected freely in the scenario that we refer to as semi non-intrusive, we explain what kind of data will be favored to address such an impact. Lastly, the semi non-intrusive viewpoint further motivates other criteria for training data, such as the training complexity, and we consider two possible types of training data that demand lower complexity.

2. BACKGROUND REVIEW

2.1. Component Forensics

We first briefly review the notion of component forensics here; for a more complete overview, please refer to [6].
Component forensics aims at identifying the algorithms and parameters employed in the various components of a device that was used to acquire the data. It provides a framework and methodology to address a number of forensics issues, including discovering device-technology infringement, protecting intellectual property rights, and identifying acquisition devices.

Depending on the available inputs to the device under question, component forensics can be considered in three main types of scenarios. In the intrusive forensics scenario, the analyst has access to the device, and can arbitrarily break the device apart to inspect each component within the device. In the semi non-intrusive forensics scenario, the analyst still has access to the device but cannot break it apart. To build forensic evidence about the components algorithms and parameters, the analyst can only design appropriate inputs to the device and examine the relation between the designed inputs and the corresponding outputs. In the completely non-intrusive forensics scenario, the analyst has no access to the device, and can only use some provided sample device outputs to estimate the component properties. It is clear that these three different scenarios correspond to different levels of forensic capabilities. While the intrusive forensics is most powerful, it may not be always available in reality. Techniques for semi and completely non-intrusive forensics thus may have higher practical values and will be the main focus of this paper.

### 2.2. Color Interpolation Coefficient Estimation

We explain the feature extraction process [2] that will be used for cell-phone camera model linkage. As illustrated in Fig. 1, light reflected from the real-world scene passes through the optical components and is then detected by an array of sensors. Most cameras in today’s consumer market employ a color filter array (CFA) to filter the lights from the scene. The CFA selectively allows a certain component of light to pass through it to the sensors. In our following discussion, let $S$ be the real-world scene to be captured by the camera and let $p$ be the CFA pattern matrix. $S(x, y, c)$ denotes a three-dimensional array (3-D) of pixel values of size $H \times W \times C$, where $H$ and $W$ represent the height and width of the image, respectively, and $C = 3$ equals to the number of color components (red, green, and blue). The CFA sampling process maps the real-world scene $S$ into a 3-D matrix $S_p$ in the form of

$$S_p(x, y, c) = \begin{cases} S(x, y, c) & \text{if } p(x, y) = c, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

After the data obtained from the CFA is available, the intermediate pixel values corresponding to the points where $S_p(x, y, c) = 0$ in (1) are interpolated using its neighboring pixel values to obtain $S_p^{(t)}$ by an operation commonly known as color interpolation or demosaicing. Following the interpolation is a post-processing stage, in which various types of in-camera processing operations such as white balancing, gamma correction, and compression may be performed to enhance the overall picture quality and/or to reduce storage space. The final camera output $S_d$ will be the overall result.

Swaminathan et al. [2] developed an algorithm to jointly estimate the CFA pattern and the color interpolation coefficients. For every possible RGB-type CFA pattern $p$ with a fixed $2 \times 2$ periodicity, the interpolation coefficients are estimated separately in different types of texture regions by linear model fitting. In specific, define $I_{x,y} = S_d(x, y, p(x, y))$, i.e., the sensor value at location $(x, y)$, then the horizontal and vertical gradients can be found as

$$H_{x,y} = |I_{x,y-2} + I_{x,y+2} - 2I_{x,y}|, \quad (2)$$
$$V_{x,y} = |I_{x-2,y} + I_{x+2,y} - 2I_{x,y}|. \quad (3)$$

Each pixel at location $(x, y)$ is classified into one of three texture regions. Region $R_1$ contains pixels satisfying $H_{x,y} > T$, i.e., pixels with a significant horizontal gradient. $T$ is a pre-determined threshold. Similarly, region $R_2$ contains pixels satisfying $V_{x,y} > T$, i.e., pixels with a significant vertical gradient. Pixels not belonging to $R_1$ or $R_2$ are assigned to region $R_3$, which mainly contains the pixels in smooth areas and some pixels that have diagonal gradients.

With a given CFA pattern $p$, the set of locations in each color channel of $S_d$ that are acquired directly from the sensor array can be determined. By approximating the remaining pixels to be interpolated with a set of linear equations in terms of the colors of directly-captured pixels, we can obtain a set of linear equations corresponding to each texture region ($R_1$, $R_2$, and $R_3$) in each color channel (red, green, and blue). This set of equations can be solved for the linear interpolation coefficients and the resulting interpolation error using the least-squares method. The overall optimal CFA pattern with the lowest interpolation error and its corresponding interpolation coefficient sets will be jointly determined.

The camera interpolation coefficients obtained as above can be used as a feature to identify the camera brand and model utilized to capture the image, as long as certain color interpolation characteristics are indeed extracted from the camera outputs and manifested in the coefficients.
Table 1. Cell-Phone Camera Models Used in Our Experiment

<table>
<thead>
<tr>
<th>Brand</th>
<th>Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sony Ericsson</td>
<td>W810i, W760i</td>
</tr>
<tr>
<td>LG</td>
<td>VX9700, VX8550</td>
</tr>
<tr>
<td>Apple</td>
<td>iPhone 3G</td>
</tr>
<tr>
<td>Samsung</td>
<td>SCH-i760, A707</td>
</tr>
<tr>
<td>Nokia</td>
<td>E71x</td>
</tr>
</tbody>
</table>

Fig. 2. A typical natural scene (a) and a typical man-made scene (b) in our experiment

3. CELL-PHONE CAMERA MODEL LINKAGE USING COLOR INTERPOLATION FEATURES

3.1. Accuracy of Cell-phone Camera Model Linkage

We first examine the cell-phone camera model linkage capability of the color interpolation coefficients, using 8 different cell-phone camera models as listed in Table 1. Please note that we will use “camera” and “camera model” interchangeably for simplicity. With each cell-phone camera, we take 50 images, among which 25 are from natural scenes and the other 25 are from man-made scenes. Natural scenes are scenes with natural materials, such as trees or leaves. Man-made scenes contain man-made structures and are taken indoors in our experiment here. Typical examples of these two types of scenes are shown in Fig. 2(a) and 2(b), respectively. Two blocks of size 512 × 512 are extracted from each image, and we have totally 50 natural-scene blocks and 50 man-made-scene blocks from each cell-phone camera.

From each block, the color interpolation coefficients are estimated and used as the features for camera model classification. The parameter \( T \) for texture region classification is set as 0.05 here. We use 7 × 7 filters to represent linear models associated with different texture regions in all color channels. Since there are three color channels and each channel has three different texture regions, the total number of filter coefficients equals to \( 7 \times 7 \times 3 \times 3 = 441 \). These coefficients form a 441-dimensional feature vector and we employ the probabilistic Support Vector Machine (SVM) to train a 8-class classifier. [2]. We randomly select 50 blocks from each camera for training and the remaining blocks for testing, and achieve an average accuracy of 90.53%.

Fig. 3. Testing accuracies for different combinations of training / testing data

3.2. Accuracies under Various Content Conditions

The average accuracy reported in the last section is found when natural scenes and man-made scenes are equally mixed in training and testing. Here we explicitly separate natural and man-made scenes to form different training and testing settings and observe the performance impacts. Fig. 3 shows four different training-testing data pairs and the corresponding camera model classification accuracy for different number of training image blocks. As we can see, the highest accuracy of around 99.96% is obtained when natural scenes are used both for training and testing. However, the accuracy drastically drops to only 62.94% as we still use natural scenes for training but use man-made scenes for testing. If we instead use man-made scenes for training, we can obtain an accuracy as high as 97.82% when natural scenes are tested and 86.38% when man-made scenes are tested. Such a trend is essentially consistent for all reasonably large training image numbers. It suggests that the performance of color interpolation coefficient estimation has to depend on the images being used.

Analysis of Coefficient Variance  We analyze the variances of the coefficients associated with different texture regions to understand the reason of varying classification accuracies. We find that the variances for coefficients associated with \( R_3 \) are similar regardless of the image type, (for example, the average variance values in red channel’s \( R_3 \) are 0.0011 and 0.0013 for natural scenes and man-made scenes, respectively), but for \( R_1 \) and \( R_2 \), the coefficient variances have a substantial difference between natural scenes and man-made scenes. This is demonstrated in Fig. 4, where in each sub-figure the horizontal axis denotes the index of coefficient (from 1 to 49 for a 7 × 7 filter size) and the vertical axis shows the variance. Color channels and texture regions are specified on top of each sub-figure. Variance values associated with natural scenes and man-made scenes are shown in dashed and solid
Fig. 4. Coefficient variances for different image contents. Solid lines: natural scenes; dashed lines: man-made scenes.

Fig. 5. Gradient distributions for different image contents.

The difference between the gradient distributions of natural and man-made scenes is also observed as in Fig. 5. The gradient of natural scenes has more large values compared to that of the man-made scenes, which is reasonable since man-made scenes are more “sparse”, i.e., having significant portions of smooth areas without large variations (see Fig. 2 for comparison). As a result, more pixels will be assigned to $R_1$ or $R_2$, and more equations are available when solving for their interpolation coefficients. This provides a better numerical condition to coefficient estimation using natural scenes, and therefore the estimates are more consistent and have lower variances. Conversely, the coefficient estimates associated with man-made scenes have less consistent and have higher variances.

A conceptual illustration of the coefficient distributions obtained from natural and man-made scenes based on our observations is given in Fig. 6, for a pair of camera models $C_1$ and $C_2$. The coefficient distributions associated with natural scenes are dense due to the smaller variances, so it is easier to distinguish different cameras when only natural-scene images are considered. Distributions associated with man-made scenes spread more widely; they are more likely to overlap with each other and thus the distinguishability is lower. Also, coefficient distributions associated with man-made scenes have a much better coverage surrounding the true coefficients, and can well capture the coefficient distributions associated with natural scenes. Therefore, using man-made scenes for training and natural scenes for testing has a much higher testing accuracy than the other way around.

We have to point out that a hard division of natural and man-made scenes can be difficult at times. Based on findings presented in this section, more categories of scenes will be examined in our following work to establish more comprehensive understandings of the content dependency issue.

4. SEMI NON-INTRUSIVE TRAINING

4.1. Training Data Selection for Accurate Matching

In the cell-phone camera model linkage problem, the analyst has no control over the images to be matched against the target camera, but with the camera at hand, he / she is able to specify the training process. Under the framework of component forensics, the testing process can be seen as completely non-intrusive while the training processing is semi non-intrusive. On the one hand, the completely non-intrusive nature of the testing process implies that undesired test data
can be present; on the other hand, the semi non-intrusive setting allows for specific training data design and can often be exploited to obtain higher component estimation accuracy. It is obvious that in cell-phone camera model linkage, the testing accuracy should be used as a goodness criterion for training data selection. Sec. 3 shows that this can be achieved using training data that yield necessary coefficient variations, and man-made scenes serve this purpose better. Since there is a roughly 2% accuracy drop if we only use man-made scenes in training to test natural scenes, and only a small number of training image blocks is required to capture the coefficients associated natural scenes (see Fig. 3), a few natural scenes are also suggested to be included in the training image set.

Note that the principle of training data variation can be applied to a broad range of problems, since the complete avoidance of content dependency and testing data variation is very difficult, if not impossible.

4.2. Training Complexity as a Training Criterion

The notion of semi non-intrusive training further provides motivations for other possible goodness criteria for training. In specific, in this paper, we consider the training complexity as one important constraint of the training procedure. It is often neglected that in the entire training process, the collection of training data may actually demand the vast majority of time than the execution of the automated machine-learning process. However, in applications particularly those in the field, a strict constraint is often imposed on the time assured for training. Although thorough capturing of representative real-world scenes, as shown in the previous section, has been shown to yield high testing accuracy, the complexity of its training data collection may be higher than affordable. In this section, we will consider some possible options for achieving a good balance between these two criteria.

Unique Issues with Cell-phone Cameras Cell-phone cameras have stricter size and functionality constraints. The majority of cell-phone cameras to date still only have fixed-focus lenses, and the sharpness of the formed image naturally decreases as the distance between the object and the camera lens reduces. Besides, cell-phone cameras usually have much weaker resilience to noise, which becomes more prominent when the lens-object distance is small and the image appears smoother. Overall, it is more challenging to use cell-phone cameras to capture images at short distances.

A low-complexity training process may require to collect pre-determined training objects in one place for training image capture, which may put implicit constraints on the relation between the object dimension and its distance to the lens. When short-distance capture becomes difficult, we suggest the use of larger objects so that the image size can be retained while keeping a larger distance between the object and lens. Besides, to facilitate image capture, planar objects will be favored.

4.3. Training Using Large Planar Objects

To examine the feasibility of using larger planar objects for training, we have considered one such object, the carpet, in our current study. A Nokia 6650d is adopted as the target camera. We collect about 15 different carpets of various textures, and take 100 image blocks of size $512 \times 512$ from these carpets as the training data, at a small distance (less than 1 feet) and a large distance (greater than 3 feet), respectively. Typical obtained carpet images are given in Fig. 7. We train a SVM-based binary classifier using another set of images taken by the target camera consisting of representative natural and man-made scene images (as the positive set) and images from other cameras as listed in Table 1 (as the negative set). The testing accuracy is evaluated in terms of the detection probability and false alarm probability, which are defined as the probability that an image taken by the target camera is correctly identified, and the probability that an image from another camera is identified as the target camera’s output, respectively. Fifty training image blocks are randomly selected and the remaining 50 are used for testing.

We find that when images taken at short distances are used for training, the detection probability is close to 0. This is due to the fact mentioned earlier that the cell-phone camera tends to generate images with reduced sharpness at a short distance, and the noise contamination becomes more severe, as shown in Fig. 7(e). These problems prevent accurate texture region classification and lead to inaccurate coefficient estimation. Nonetheless, when images taken at larger distances are used instead, the detection probability for man-made scene images and the natural scene images are about 60% and close to 100%, respectively, while the false alarm probability is still kept as low as 3.7%. These observations suggest that it is possible to design a more controlled and low-complexity training procedure which still provides good testing accuracy.

4.4. Training Using Displayed Images

We have also begun to consider another possible low-complexity training scenario by displaying sample images on LCD screen as training objects. These sample images are usually first acquired by other high-end cameras to avoid up-sampling when displayed on a LCD screen. We use the same Nokia 6650d model as the target camera and generate the sample images containing both natural and man-made scenes using a Canon PowerShot G7 standalone camera with resolution $3648 \times 2736$. The used LCD resolution is $1280 \times 1024$.

For the same cell-phone camera Nokia 6650d, using such recaptured images for training only yields a detection probability close to 0. To see whether an improved short-distance capture capability may help, we use another camera, a Canon PowerShot SD800 IS with macro mode as the target camera and redo the experiment. The detection probabilities for natural-scene images and man-made scenes are 10.05% and 38.08%, respectively. While such accuracies are still
Fig. 7. (a) and (b): Carpet images captured at a larger distance; (c) and (d): carpet images captured at a larger distance. (e): Local patch of a carpet image taken at a close distance; note the lower sharpness and higher noise level.

Fig. 8. Four consider training objects placed on the Training Criteria Plane

far lower than those achieved using representative real-world scenes or large planar objects, there is a clear improvement due to the camera’s short-distance capture capability.

4.5. Discussions

As the design of different training procedures is still in its early stage, issues remain in the above two training methods. For the training using large planar objects such as textiles or carpets, the required variations in the texture may be limited as compared to real-world scenes, and therefore the achievable testing accuracy may be lower. Also, such large planar objects may not be readily available in practical use. When using displayed images for training, besides the sharpness reduction we report in the planar-object experiment, it has been also found in recent literature [7] that recaptured LCD images have additional textures from the display and recapturing process. Such textures may bias the coefficient estimation.

By considering testing accuracy and training complexity explicitly, we propose and plot the Training Criteria Plane as in Fig. 8, placing the three types of training data we have tried on the corresponding positions. The ultimate goal is to reach the bottom-right corner of the plane. One direction is to investigate if we can construct artificial patterns that yield high testing accuracy and can be captured easily.

5. CONCLUSIONS

In this paper, we first present a study of cell-phone camera model linkage using color interpolation coefficient features. A detailed analysis of the achievable performance with respect to different image contents shows a non-negligible amount of content dependency. Since the training process is semi non-intrusive under the framework of component forensics, a proper selection of training data that provide sufficient variations in the coefficient estimates can be employed to deal with the content dependency and improve the testing accuracy. The viewpoint of semi non-intrusive training has also motivated other alternative training data when different goodness criteria are applied. We consider the training complexity in this paper and have shown that some training data such as large planar objects and displayed images can serve as training objects but have reduced forensic performance.

6. REFERENCES