INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Detecting Photograph Manipulation Through Color Constancy Analysis and Feature Engineering

Tiago Carvalho *Christian Riess*
Elli Angelopoulou *Helio Pedrini*
Anderson Rocha

Technical Report - IC-12-25 - Relatório Técnico

October - 2012 - Outubro

Detecting Photograph Manipulation Through Color Constancy Analysis and Feature Engineering

Tiago Carvalho* Christian Riess† Elli Angelopoulou Helio Pedrini
Anderson Rocha

Abstract

For decades, photographs were taken as a unique form of documenting space-time events. As such they have often served as evidence in courts. Although, in principle, photographers are able to create analog composites, e. g.,for comic or dramatic effects, this process is very time consuming and requires expert knowledge. However, nowadays, powerful digital image editing software packages have made image modifications easier than ever. This undermines our trust in images and, in particular, questions pictures as evidence for real-world events. In this context, here we analyze one of the most common forms of photograph manipulation nowadays, known as image composition or splicing in which multiple images are combined to create a new, fake photograph. We propose a forgery detection method that exploits subtle inconsistencies in the color of the illumination of images. Our approach is machine learning-based and requires minimal user interaction. In contrast to prior work, the method is applicable to a broad range of images and requires no expert interaction for the tampering decision. To achieve this, we incorporate cues from physics- and statistical-based illuminant estimators on image regions of similar material. From these illuminant estimates we extract texture- and edge-based features feeding a machine learning approach for automatic decision-making. The classification performance using an SVM meta-fusion classifier is promising, yielding a tampering detection rate of 85% on a new image forensics benchmark of 100 skillfully created forgeries and 100 pristine photographs.

1 Introduction

Every day, millions of digital documents are produced by a variety of devices and distributed by newspapers, magazines, websites and television. In all these information channels, images are a powerful tool for communication. Unfortunately, it is not difficult to use computer graphics and image processing techniques to manipulate images, forging new realities, and diminishing people's trust in images. Quoting Russell Frank, a Professor of Journalism Ethics at Penn State University, back in 2003 right after a Los Angeles Times incident involving a doctored photograph from the Iraqi front: "Whoever said the camera never lies

[*] RECOD Lab, Institute of Computing, University of Campinas, Brazil.Research conducted with partial financial support from CNPq, process 140916/2012-1

[†] Pattern Recognition Lab, University of Erlangen-Nuremberg, Germany

was a liar". Frank could not be more accurate in his analysis. While we have relied on photographs for decades as a truthful means of information, it turns out that the technology era has made it easier than ever to manipulate photographs and forge different realities.

How we deal with such technology towards photograph manipulation raises a surprising host of legal and ethical questions that we must address [1]. When does image manipulation turn from simple tweaking of family photos to more serious, potentially criminal, manipulation of the public opinion? Before one can think of taking appropriate actions upon a questionable image, one must be able to detect that an image has been altered [1].

As a matter of fact, image composition (or splicing) is one of the most common image manipulation operations nowadays. Here, parts from two or more source images are used to compose a new image that allegedly testifies a real-world event. People are a common subject of spliced images. One such example is the advertisement of the "Unhate" campaign, created by the Benetton Agency (see Figure 1), which depicts two heads of state kissing each other.

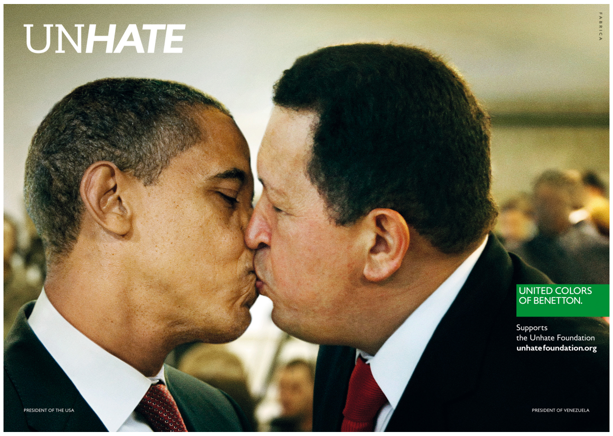


Figure 1: How can one assure the authenticity of a picture? Example of a spliced image involving people within an advertising campaign. Picture courtesy of Benetton Group 2011 (http://press.benettongroup.com/).

To use an image as a trusted document for a particular event that actually happened, forensic investigators try to detect all possible tampering telltale signs in a given image in order to expose image forgeries. Manipulation cues are, among others, compression artifacts, natural image statistics, image acquisition artifacts, and illumination inconsistencies [2, 1]. Methods based on illumination inconsistencies have two main characteristics that make them potentially effective in splicing detection. Firstly, from the viewpoint of a manipulator, a perfect adjustment of illumination conditions is very difficult to achieve when creating a composite photograph. Secondly, this class of methods can also be used to analyze analog pictures, as discussed, for instance, in [3].

Illumination color analysis is a promising cue to expose image composites. In earlier work, Riess and Angelopoulou [4] proposed to analyze illuminant color estimates from local image regions to detect spliced images. Unfortunately, the authors leave the interpretation of the so-called illuminant maps to human experts. In practice, it turns out that it is very challenging to decide whether or not an image is tampered with based just on illuminant maps. Moreover, we can not simply rely solely on a subject's or expert's opinion, as the human visual system can be quite inept at judging inconsistencies in photographs, especially when it involves lighting and shadows [5, 6]. Thus, it is necessary to transfer the tampering decision to an objective automated algorithm as much as possible.

In this work, we make an important step towards minimizing user interaction and human expertise in analyzing and interpreting such illuminant maps. We propose a new semi-automatic method that is considerably easier to use and more reliable than earlier approaches. We make use of the fact that local illuminant estimates are most discriminative when comparing objects that are made of the same (or similar) material. Thus, we focus on the automated comparison of regions of human skin, and more specifically, faces. We classify the illumination on two faces as either consistent or incosistent. The only interaction that is required by the user is to select image regions that contain objects of similar materials. Specifically, we restrict the required user interaction to marking bounding boxes around the faces in an image under investigation. In the simplest case, this reduces to specifying two corners (upper left and lower right) of the bounding box.

The main contributions of this work are:

- The combination of multiple image illuminant maps using physics- and a statistical-based color constancy estimation methods towards a more reliable authenticity analysis.
- A novel edge-based characterization method for illuminant maps.
- Interpretation of illuminant information as texture maps and further characterization of such maps through texture analysis methods.
- The creation of a photorealistic image benchmark comprising 100 skillfully created forgeries and 100 photographs for standardization of image composition analysis[1].
- The development of a machine learning fusion method which incorporates complementary features of distinct illuminant maps for the final decision-making process yielding a detection rate of about 85%.

Finally, we organize this paper into five sections: in Section 2, we briefly review related work in color constancy and illumination-based detection of image splicing. In Section 3, we present an overview of the methodology, followed by a detailed explanation of all the algorithmic steps. In Section 4, we describe the proposed benchmark database, and experimental results. Section 5 concludes the paper and hints at potential future work.

[1] The dataset will be available in full two mega pixel resolution upon the acceptance of the paper. For reference, all images in lower resolution can be viewed at: http://www.ic.unicamp.br/~tjose/files/database-tifs-small-resolution.zip

2 Related Work

Illumination-based methods for forgery detection are either geometry-based or color-based. Geometry-based methods aim at detecting inconsistencies in light source positions between specific objects in the scene [3, 7, 8, 9, 10, 11]. Color-based methods search for inconsistencies in the interactions between object color and light color [12, 4].

Two methods have been proposed to use the direction of the incident light for exposing digital forgeries. Johnson and Farid [8] proposed a method which computes a low-dimensional descriptor of the lighting environment in the image plane (i. e., in 2D). It estimates the illumination direction from the intensity distribution along manually annotated object boundaries of homogeneous color. Kee and Farid [10] extended this approach to additionally explore known 3D surfaces. The authors demonstrate, for the case of faces, that a dense grid of 3D normals can improve the estimate of the illumination direction. To achieve this, a 3D face model is registered with the 2D image using manually annotated facial landmarks.

Johnson and Farid [9] also proposed solutions for special cases. For instance, to investigate images containing spliced people, they proposed a method for detecting forgeries using specular highlights in the eyes. Saboia et al. [13] automatically classified these images by extracting additional features, such as the viewer position. The applicability of both approaches, however, is somewhat limited in practice by the fact that people's eyes must be visible and available in high resolution.

Gholap and Bora [12] introduced physics-based illumination cues to image forensics. The authors examined inconsistencies in specularities based on the dichromatic reflection model. Specularity segmentation on real-world images is challenging [14]. Therefore, the authors require manual annotation of specular highlights. A second drawback of this approach is that it relies on the presence of specularities on all regions of interest making them difficult to deploy in many real-world scenarios. To avoid this problem, Wu and Fang [15] assume purely diffuse reflectance (i.e., scenes without specularities), and train a mixture of Gaussians to select a proper illuminant color estimator. The angular distance between illuminant estimates from selected regions can then be used as an indicator for tampering. Unfortunately, the authors require the manual selection of a "reference block", where the color of the illuminant is estimated with sufficient accuracy. This restricts the applicability of the method to scenes containing favorable background, and the selection requires a human expert.

Riess and Angelopoulou [4] followed a different approach, using a physics-based color constancy algorithm that operates on partially specular pixels. In this approach, the segmentation of pure specularity is avoided. The authors propose to segment the image in homogeneous regions and estimate the illuminant color *per segment* (i.e., locally). Recoloring the image per segment with local estimates yields the so-called *illuminant maps*. Implausible illuminant color estimates point towards a manipulated region. Unfortunately, the authors do not provide a numerical decision criterion for tampering detection. Thus, an expert is left with the difficult task of visually examining an illuminant map for evidence of tampering.

In this paper, we build upon these ideas. We use the relatively rich illumination information provided by physics-based and statistical color constancy methods as in [4, 16], which we explore for an improved, semi-automated and accurate detection of image forgeries. Decisions with respect to the illuminant color estimators are completely taken away from the user, which differentiates this work from [4] and [15].

Most research in color constancy focuses on uniformly illuminated scenes containing a single dominant illuminant. For an overview, see [17, 18, 19]. As a cue for image forensics, however, we require multiple, spatially-bound estimates for the color of the illuminant. So far, limited research has been done in this direction. The work by Bleier et al. [20] indicates that many off-the-shelf single-illuminant algorithms do not scale well on smaller image regions. As a consequence, a careful, problem-specific selection of a method for illuminant estimation is required.

Besides the work of [4], Gijsenij et al. [21] proposed a pixelwise illuminant estimator to segment an image into regions illuminated by distinct illuminants. The pixelwise estimate allows crisp transitions between differently illuminated regions, for instance between sunlit and shadow areas. While this is an interesting approach, a single illuminant estimator can always fail. Thus, for forensic purposes, we prefer a scheme that fuses multiple illuminant estimates. Earlier, Kawakami et al. [22] proposed a physics-based approach that is tailored for discriminating shadow/sunlit regions. However, for our work, we consider the restriction to outdoor images overly limiting. Ebner [23] presented a more general approach to multi-illuminant estimation. Assuming smoothly blending illuminants, the author proposes a diffusion process to recover the colors of the illuminants. Unfortunately, in practice, this approach tends to oversmooth the illuminant boundaries, and thus does not leave sufficiently detailed information in the illuminant estimates.

Each of the previously mentioned methods have their individual failure cases or limitations. Thus, to improve the robustness, here we *combine* two methods that operate on different principles. In detail, we used estimates from a statistical method (by van de Weijer et al. [16]) and a physics-based method (by Riess and Angelopoulou [4]).

3 Core of the System: Estimation of the Locally Dominant Illuminant and Interpretation of Illuminant Maps

In Section 2, we presented efforts, based on illumination inconsistencies, that forensic researchers are employing to fight against this kind of forgeries. However, there are still very few such methods found in the literature, mainly because they are usually dependent on user experience.

To overcome this limitation, we propose a new approach that minimizes user dependence and improves the state-of-the-art. We classify the illumination for each pair of faces in the image as either consistent or inconsistent. First, we present an overview of the algorithm. Then, we present the algorithmic details for every step. Throughout the paper, we abbreviate illuminant estimation as IE, and illuminant maps as IM. The proposed method consists of five main components:

1. *Dense Local Illuminant Estimation (IE):* the input image is segmented into homogeneous regions. Per illuminant estimator, a new image is created where each region is

colored with the estimated illuminant color. This resulting intermediate representation is called illuminant map (IM).

2. *Face Extraction:* this is the only step that may require human interaction. An operator sets a bounding box around each face (e. g., by clicking on two corners of the bounding box) in the image that should be investigated. Alternatively, an automated face detector can be employed. We then crop every bounding box out of each illuminant map, such that only the illuminant estimates of the face regions remain.

3. *Computation of Illuminant Features:* for all face regions, texture-based and gradient-based features are computed on the IM values. Further analysis is performed on these features.

4. *Paired Face Features:* our goal is to assess whether two faces in an image are consistently illuminated. For that, we combine the feature vectors from each pair of faces in the image creating a pair-of-faces feature vector. For an image with f faces, we construct a combination of $\binom{f}{2}$ feature vectors.

5. *Classification:* we use a machine learning approach to automatically classify the feature vectors. Given an image with f faces, we consider an image as a forgery if at least one pair of faces (represented by one feature vector) is classified as inconsistently illuminated.

Figure 2 summarizes the main steps of the proposed method. The remainder of this section presents the details of such steps.

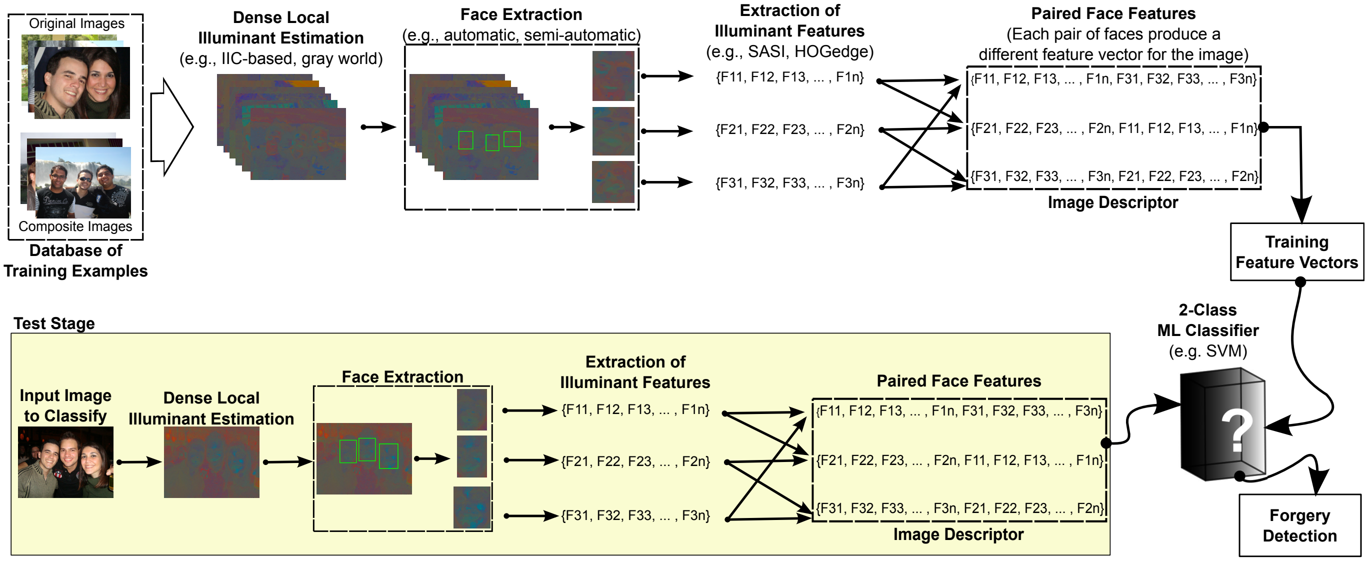

Original Images
Composite Images
Database of Training Examples
Dense Local Illuminant Estimation (e.g., IIC-based, gray world)
Face Extraction (e.g., automatic, semi-automatic)
Extraction of Illuminant Features (e.g., SASI, HOGedge)
{F11, F12, F13, ... , F1n}
{F21, F22, F23, ... , F2n}
{F31, F32, F33, ... , F3n}
Paired Face Features (Each pair of faces produce a different feature vector for the image)
{F11, F12, F13, ... , F1n, F31, F32, F33, ... , F3n}
{F21, F22, F23, ... , F2n, F11, F12, F13, ... , F1n}
{F31, F32, F33, ... , F3n, F21, F22, F23, ... , F2n}
Image Descriptor
Training Feature Vectors
Test Stage
Input Image to Classify
Dense Local Illuminant Estimation
Face Extraction
Extraction of Illuminant Features
{F11, F12, F13, ... , F1n}
{F21, F22, F23, ... , F2n}
{F31, F32, F33, ... , F3n}
Paired Face Features
{F11, F12, F13, ... , F1n, F31, F32, F33, ... , F3n}
{F21, F22, F23, ... , F2n, F11, F12, F13, ... , F1n}
{F31, F32, F33, ... , F3n, F21, F22, F23, ... , F2n}
Image Descriptor
2-Class ML Classifier (e.g. SVM)
?
Forgery Detection

Figure 2: Overview of the proposed method.

3.1 Dense Local Illuminant Estimation

To detect inconsistencies in the illumination color, we need a dense set of localized estimates. We segment the input image into regions of approximately constant chromaticity (so-called superpixels) with the algorithm proposed by Felzenszwalb and Huttenlocher [24]. Then we estimate the color of the illuminant per superpixel. By recoloring the superpixels with the estimated illuminant chromaticities, we obtain an illuminant map. We use two separate methods to obtain a version of this map: the statistical generalized gray world estimates and the physics-based inverse-intensity chromaticity space, as we explain next.

3.1.1 Generalized Gray World Estimates

The classical gray world assumption by Buchsbaum [25] states that the average color of a scene is gray. Thus, a deviation of the average of the image intensities from the expected gray color is due to the illuminant. Although this assumption is nowadays considered to be overly simplified [18], it has inspired the further design of statistical descriptors for color constancy. We follow an extension of this idea, the generalized gray world approach by van de Weijer et al. [16].

Let $\mathbf{f} = (R, G, B)^{\mathrm{T}}$ denote the observed color of a pixel. Van de Weijer et al. [16] assume a Lambertian scene (i. e., objects of purely diffuse reflectance) and linear camera response. Then, $\mathbf{f}$ is formed by

$$\mathbf{f} = \int_{\Omega} e(\lambda)s(\lambda)\mathbf{c}(\lambda)d\lambda \ , \tag{1}$$

where λ denotes the wavelength of the light, $e(\lambda)$ denotes the spectrum of the illuminant, $s(\lambda)$ the surface reflectance of an object, and $\mathbf{c}(\lambda)$ the sensitivity of the camera as a vector for each color channel. Van de Weijer [16] et al. extended upon the original gray world hypothesis through the incorporation of three parameters:

1. Derivative order n: the assumption that the average of the illuminants is achromatic can be extended to the absolute value of the sum of the derivatives of the image.

2. Minkowski norm p: instead of simply adding intensities or derivatives, respectively, greater robustness can be achieved by computing the p-th Minkowski norm of these values.

3. Gaussian smoothing σ: to reduce image noise, one can smooth the image prior to processing with a Gaussian kernel of standard deviation σ.

Putting these three aspects together, van de Weijer et al. proposed to estimate the color of the illuminant $\mathbf{e}$ as

$$k\mathbf{e}^{n,p,\sigma} = \left(\int \left| \frac{\partial^n \mathbf{f}^{\sigma}(\mathbf{x})}{\partial \mathbf{x}^n} \right|^p d\mathbf{x} \right)^{1/p} \quad , \tag{2}$$

where k denotes a scaling factor, $|\cdot|$ the absolute value, ∂ the differential operator, and $\mathbf{f}^{\sigma}(\mathbf{x})$ the observed intensities at position $\mathbf{x}$, smoothed with a Gaussian kernel σ. Note that $\mathbf{e}$ can be computed separately for each color channel. Compared to the original gray world algorithm, the derivative operator increases the robustness against homogeneously colored regions of varying sizes. Additionally, the Minkovski norm emphasizes strong derivatives over weaker derivatives, so that specular edges are better exploited [26].

3.1.2 Inverse Intensity-Chromaticity Estimates

The second illuminant estimator we consider in this paper is the so-called inverse intensity-chromaticity (IIC) color space. It was originally proposed by Tan et al. [27]. In contrast to the previous approach, the observed image intensities are assumed to exhibit a mixture of diffuse (i. e., Lambertian) and specular reflectance. Pure specularities are assumed to consist of only the color of the illuminant. Let (as above) $\mathbf{f} = (R, G, B)^{\mathrm{T}}$ be the observed colors of a pixel. Then, using the same notation as for the generalized gray world model, $\mathbf{f}$ is modelled as

$$\mathbf{f} = \int_{\Omega} (e(\lambda)s(\lambda) + e(\lambda))\mathbf{c}(\lambda)d\lambda \ . \tag{3}$$

Let $\mathbf{f}_c(\mathbf{x})$ be the intensity and $\chi_c(\mathbf{x})$ be the chromaticity (i. e., normalized RGB-value) of a color channel c at position $\mathbf{x}$, respectively. In addition, let γ_c be the chromaticity of the illuminant in channel c. Then, after a somewhat laborious calculation, Tan et al. [27] derived a linear relationship between $\mathbf{f}$, χ_c and γ_c by showing that

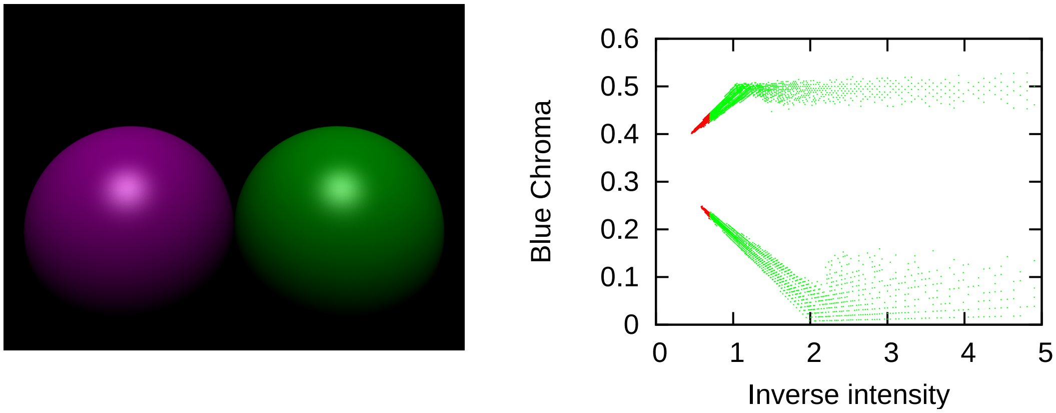
$$\chi_c(\mathbf{x}) = m(\mathbf{x}) \frac{1}{\sum_i \mathbf{f}(\mathbf{x})} + \gamma_c \ . \tag{4}$$

Here, $m(\mathbf{x})$ mainly captures geometric influences, i. e., light position, surface orientation and camera position. Although $m(\mathbf{x})$ can not be computed analytically, an approximate solution is feasible. More importantly, the only aspect of interest in illuminant color estimation is the y-intercept γ_c. This can be directly estimated by analyzing the distribution of pixels in IIC space. The IIC space is a per-channel 2D space, where the horizontal axis is the inverse of the sum of the chromaticities per pixel, $1/\sum_i \mathbf{f}_i$, and the vertical axis is the pixels's chromaticity for that particular channel. Per color channel c, the pixels within a superpixel are projected onto Inverse Intensity-Chromaticity (IIC) space.

Figure 3 depicts an exemplary IIC diagram for the blue channel. A synthetic image is rendered (left) and projected onto IIC space (right). Pixels from the green and purple balls form two clusters. The clusters have spikes that point towards the same location at the y-axis. Considering only such spikes from each cluster, the illuminant chromaticity is estimated from the joint y-axis intercept of all spikes in IIC space [27].

In natural images, noise dominates the IIC diagrams. Riess and Angelopoulou [4] proposed to sample small regions of interest and perform principal component analysis (PCA) for each region in IIC space. Let λ_1 and λ_2 denote the largest and second largest eigenvalues that result from the PCA. To select the spikes in IIC space, two conditions are checked. First, the eccentricity [2] (i. e., $\sqrt{1 - \sqrt{\lambda_2}/\sqrt{\lambda_1}}$) must exceed a minimum of 0.2. Second, the slope [2] of $\mathbf{v}_1$ must be higher in absolute terms than 0.003. Then, a single γ_c is estimated as

[2] Such parameter values were previously investigated by Riess and Angelopoulou [4, 28] through a battery of experiments. In this paper, we rely on their findings.



(a)

(b)

Figure 3: Illustration of the inverse intensity-chromaticity space (blue color channel). Left: synthetic image (violet and green balls). Right: specular pixels converge towards to the blue portion of the illuminant color (recovered at the y-axis intercept). Highly specular pixels are shown in red.

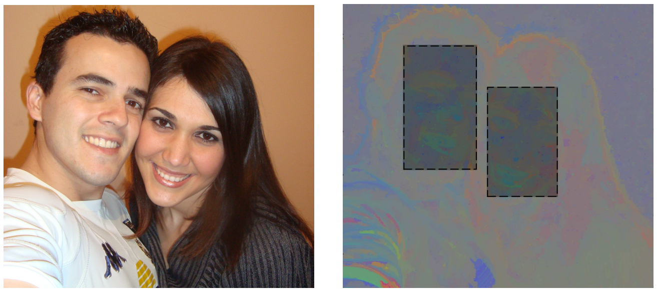
the intersection with the y axis with the eigenvector $\mathbf{v}_1$ corresponding to λ_1. Finally, the color of the illuminant is determined via a consensus vote over all the regions of interest in a segment.

3.2 Face Extraction

Unconstrained estimation of the illuminant color can be error-prone and affected by the reflectance properties of the materials in the scene. However, it is possible to improve the accuracy of the relative error between two estimates by focusing only on objects of approximately the same material. For this work, we limit our examination of illumination consistency to human skin and, in particular, to faces. Pigmentation is the most obvious difference in skin characteristics between different ethnicities. This pigmentation difference depends on many factors as quantity of melanin, amount of UV exposure, genetics, melanosome content and type of pigments found in the skin [29]. However, this intra-material variation is typically smaller than that of all materials possibly occurring in a scene.

All faces in the image that should be part of the investigation have to be annotated with a bounding box. In principle, this can be done automatically, through the use of a face detector [30]. However, we prefer a human operator for this task for two main reasons: a) this minimizes false detections or misses of faces; b) scene context is important when judging the lighting situation. For instance, consider an image where all persons of interest are illuminated by flashlight. The illuminants are expected to agree with one another. Conversely, assume that a person in the foreground is illuminated by flashlight, and a person in the background is illuminated by ambient light. Then, a difference in the color of the illuminants is expected. Such differences are hard to distinguish in a fully-automated manner. For this paper, we focused on faces that are exposed to supposedly similar illumination, which can be visually verified by the operator.

We illustrate this setup in Figure 3.2. The faces in Figure 4(a) can be assumed to be exposed to the same illuminant. As Figure 4(b) shows, the corresponding gray world illuminant map for these two faces also has similar values.



(a) Original

(b) Gray world with similar parts highlighted

Figure 4: An original image and its gray world map. Highlighted regions in the gray world map show a similar appearance.

3.3 Interpreting Illuminant Maps as Texture Maps

From an image processing perspective, we can interpret the illuminant maps from face regions as texture maps. Many different texture descriptors have been proposed in the literature thus far. One of the most effective [31] is the Statistical Analysis of Structural Information (SASI) [32] descriptor. The most important advantage of SASI for our application is its remarkable capability of capturing small granularities and discontinuities which are present in texture patterns. These patterns appear mainly in sharp corners and abrupt changes such as the ones present in illuminant maps, especially in the face region of composite images.

3.3.1 SASI

SASI [32] is a generic descriptor that measures the structural properties of texture. It captures these properties using sliding windows through the image. Given that each window is composed by a different orientation and resolution, a *clique window* W_c represents a c-nth window with a specific orientation and resolution.

When sliding through an image, every clique window uses a specific orientation determined by a *lag vector* defined as

$$\bar{v} = (v_x, v_y) \tag{5}$$

where v_x and v_y represents two different locations into a clique window. For each clique window W_c, the algorithm traverses the image calculating an autocorrelation coefficient as Equation 6 shows. In the end, every clique window W_c produces a different autocorrelation

$$r\left(\bar{v}\right)^{W_c} = \frac{\sum\limits_{\forall (i,j) \ \text{and} \left(i+v_x, j+v_y\right) \in W_c} \left(x_{i,j} - \bar{x}_{i,j}\right)\left(x_{i+v_x, j+v_j} - \bar{x}_{i+v_x, j+v_j}\right)}{\sqrt{\sum\limits_{\forall (i,j) \in W_c} \left(x_{i,j} - \bar{x}_{i,j}\right)^2 \sum\limits_{\forall \left(i+v_x, j+v_y\right) \in W_c} \left(x_{i+v_x, j+v_j} - \bar{x}_{i+v_x, j+v_j}\right)^2}} \tag{6}$$

In this equation, $x_{i,j}$ is the image gray value at position (i, j) and $\bar{x}_{i,j}$ is the mean gray value of clique window W_c.

image. We use the mean and standard deviation extracted from all autocorrelation images to compose the final image feature vector.

3.3.2 HOGedge: A New Algorithm for Interpreting Edges of Illuminant Maps

When a person from another image is used to create a spliced forgery, local discontinuities in the illuminant maps may occur. These discontinuities affect mainly the edges of an illuminant map at the splicing boundary. To characterize this local information, we propose a new algorithm named *HOGedge*. It is based on the well-known HOG-descriptor, and computes visual dictionaries of gradient intensities in edge points. The full algorithm is described in the remainder of this section. Figure 5 shows an algorithmic overview of the method.

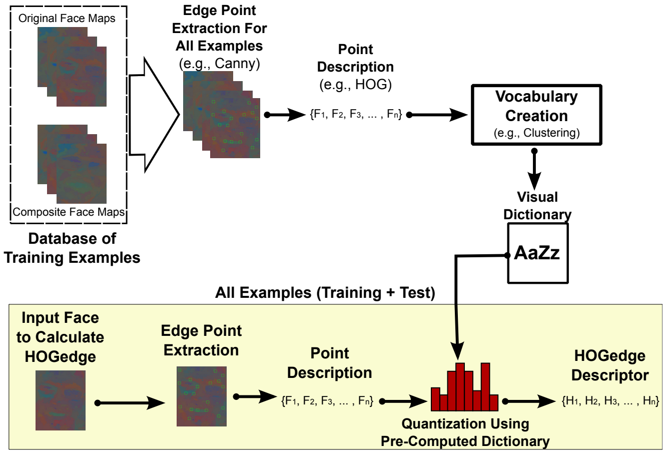
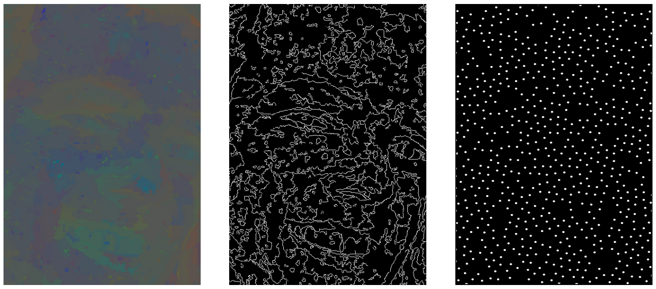

Original Face Maps
Composite Face Maps
Database of Training Examples
Edge Point Extraction For All Examples (e.g., Canny)
Point Description (e.g., HOG)
{F1, F2, F3, ... , Fn}
Vocabulary Creation (e.g., Clustering)
Visual Dictionary
AaZz
All Examples (Training + Test)
Input Face to Calculate HOGedge
Edge Point Extraction
Point Description
{F1, F2, F3, ... , Fn}
Quantization Using Pre-Computed Dictionary
HOGedge Descriptor
{H1, H2, H3, ... , Hn}

Figure 5: Overview of the proposed HOGedge algorithm.

HOGedge Algorithm The algorithm for characterizing a face using HOGedge descriptor is divided into two parts. First, we construct a visual dictionary using training examples (faces representing normal and doctored examples from illuminant maps). Then, we construct the final feature vector for every image in a dataset using the learned visual dictionary.

Extraction of Edge Points Given a face region from an illuminant map, we first extract edge points using the Canny edge detector [33]. However, this produces a large number of spatially close edge points. To reduce the number of points,we filter the Canny output using the following rule: starting from a seed point, we eliminate all other edge pixels in a region of interest (ROI) centered around the seed point. The edge points that are closest to the ROI (but outside of it) are chosen as seed points in the next iteration. Repeating this process for the whole image, we reduce the number of points in a face by excluding points that are too close to each other but without ensuring that every face has the same number of points. Figure 6 depicts an example of the resulting points.



(a) IM derived from gray world (b) Canny Edges (c) Filtered Points

Figure 6: (a) The gray world IM for the left face in Figure 4(a). (b) The result of the Canny edge detector when applied on this IM. (c) The final edge points after filtering using a square region.

Point Description We compute Histograms of Oriented Gradients (HOG) [34] to describe the edge points. HOG is based on evaluating normalized local histograms of image gradient orientations in a dense grid. The HOG descriptor is constructed by dividing the image, or region of interest, into small spatial regions ("cells"). Each cell provides a local 1-D histogram of quantized gradient directions (or edge orientations) using all cell pixels. To construct the final feature vector, the histograms of all cells in a somewhat larger spatial regions (the "blocks") are combined and contrast-normalized using an accumulated measure of local histograms (the "energy"). We use the HOG output as a feature vector for the subsequent steps.

Visual Vocabulary The number of edge points, and therefore the number of HOG vectors, varies depending on the face under examination. To obtain one unified feature vector, we use the concept of visual dictionaries [35]. Visual dictionaries constitute a robust representation, such that each face is treated as a collection of regions. The only information of interest is the appearance of each region [36].

To construct our visual dictionary, we first need to construct two sets of feature vectors: one from original and one from doctored images, using training data. Clustering each set with n cluster centers using the k-means algorithm [37], we compose a visual dictionary with $2n$ visual words, where each word is represented by a cluster center. Thus, the visual dictionary comprises the most representative feature vectors of the training set. Algorithm 1 shows the pseudocode for the dictionary creation.

Algorithm 1 HOGedge – Visual dictionary creation

```
Input: $V_{TR}$ (training database examples)
       $n$ (the number of visual words by class)
Output: $V_D$ (visual dictionary containing $2n$ visual words)
  $V_D \leftarrow \emptyset$;
  $V_{NF} \leftarrow \emptyset$;
  $V_{DF} \leftarrow \emptyset$;
  for each face $i \in V_{TR}$ do
    $V_{EP} \leftarrow$ edge points extracted from $i$;
    for each point $j \in V_{EP}$ do
      $FV \leftarrow$ apply HOG in image $i$ at position $j$;
      if $i$ is a doctored face then
        $V_{DF} \leftarrow \{V_{DF} \cup FV\}$;
      else
        $V_{NF} \leftarrow \{V_{NF} \cup FV\}$;
      end if
    end for
  end for
  Cluster $V_{DF}$ using $n$ centers;
  Cluster $V_{NF}$ using $n$ centers;
  $V_D \leftarrow \{$centers of $V_{DF} \cup$ centers of $V_{NF}\}$;
  return $V_D$;
```

Quantization Using the Pre-Computed Visual Dictionary For evaluating feature vectors, the HOG feature vectors are mapped to the visual dictionary. Each feature vector in an image is represented by the closest word (with respect to the Euclidean distance) in the dictionary. A histogram of word-counts represents the distribution of feature vectors in a face. Algorithm 2 shows the pseudocode for the application of the visual dictionary on IMs.

3.4 Face Pair

To compare two faces, we combine the same descriptors for each of the two faces. For instance, we can concatenate the SASI-descriptors that were computed on gray world. The idea is that a feature concatenation from two faces is different when one of the faces is an original and one is spliced. For an image containing f faces ($f \geq 2$), the number of face pairs is $(f(f-1))/2$.

3.5 Classification

We classify the illumination for each pair of faces in an image as either consistent or inconsistent. Assuming all selected faces are illuminated by the same light source, we tag an image as manipulated if one pair is classified as inconsistent. Individual feature vectors, i. e., texture or HOGedge features on either gray world or IIC-based illuminant maps, are classified using a support vector machine (SVM) classifier with a radial basis function (RBF) kernel.

Algorithm 2 HOGedge – Face characterization

```
Input: $V_D$ (visual dictionary pre-computed with $2n$ visual worlds)
       $IM$ (illuminant map from a face)
Output: $HFV$ (HOGedge feature vector)
  $HFV \leftarrow 2n$-dimensional vector, initialized to all zeros;
  $V_{FV} \leftarrow \emptyset$;
  $V_{EP} \leftarrow$ edge points extracted from $IM$;
  for each point $i \in V_{EP}$ do
    $FV \leftarrow$ apply HOG in image $IM$ at position $j$;
    $V_{FV} \leftarrow \{V_{FV} \cup FV\}$;
  end for
  for each feature vector $i \in V_{FV}$ do
    $\mathit{lower\_distance} \leftarrow +\infty$;
    $\mathit{position} \leftarrow -1$;
    for each visual word $j \in V_D$ do
      $\mathit{distance} \leftarrow$ Euclidean distance between $i$ and $j$;
      if $\mathit{distance} < \mathit{lower\_distance}$ then
        $\mathit{lower\_distance} \leftarrow \mathit{distance}$;
        $\mathit{position} \leftarrow$ position of $j$ in $V_D$;
      end if
    end for
    $HFV[\mathit{position}] \leftarrow HFV[\mathit{position}] + 1$;
  end for
  return $HFV$;
```

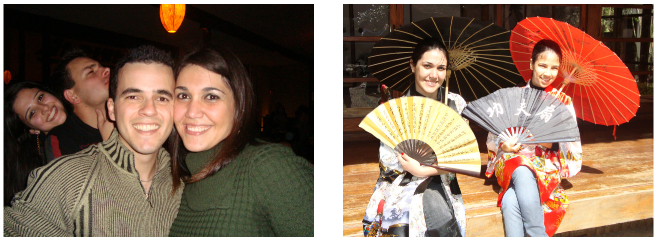
The information provided by the SASI features is complementary to the information from the HOGedge features. Thus, we use a machine learning-based fusion technique for improving the detection performance. Inspired by the work of Ludwig et al. [38], we use a late fusion technique named SVM-Meta Fusion.We classify each combination of illuminant map and feature type independently (i. e., SASI-Gray-World, SASI-IIC, HOGedge-Gray-World and HOGedge-IIC) using a two-class SVM classifier to obtain the distance between the image and the classifier decision boundary. SVM-Meta fusion consists of merging the margin distances provided by all m individual classifiers to build a new feature vector. Another SVM classifier (i. e. on meta level) then classifies the combined feature vector.

4 Experiments

To validate our approach, we performed two sets of experiments using a new database with 200 images involving people. We show results using classical ROC curves where *sensitivity* represents the number of composite images correctly classified and *specificity* represents the number of original images (non-manipulated) correctly classified.

4.1 Database

The database we created is composed of 200 indoor and outdoor images, with an image resolution of $2,048 \times 1,536$ pixels. Each image contains two or more people. From this dataset, 100 images are original, i. e., have no adjustments. The remaining 100 images are doctored. The forgeries have been composed by adding one or more people in a source image that already contained one or more people. When necessary, we performed color and image adjustments to construct photo realistic forgeries. Figure 7 depicts two example images from the dataset.



(a) Original image. (b) Spliced image.

Figure 7: An original image (left) and a spliced image (right).

4.2 Performance of Forgery Detection using Semi-Automatic Face Annotation

We compare five variants of the method discussed in this paper. Throughout this section, we manually annotated the faces using corner clicking detection (which is explained in Section 4.3). In the classification stage, we use a five-fold cross validation protocol, an SVM classifier with an RBF kernel, and classical grid search for adjusting parameters in training samples [37]. Since each image provides a different number of feature vectors, we also use a proportional weight of classes to equalize them in the training stage. Let w_o represent the number of feature vectors extracted paired faces in non-manipulated (pristine) images during training, and w_c represent the number of feature vectors extracted from paired faces of composite images also during training. To use a proportional class weighting, we simply set the weight of non-manipulated image class to $w_c/\left(w_o + w_c\right)$ and the weight of composite image class to $w_o/\left(w_o + w_c\right)$.

As for the experiments, we compared these five experimental setups:

- **SASI-IIC:** we extract SASI-features from an IIC-based illuminant map. The SASI descriptor is calculated over the Y channel from the YC_bC_r color space. The SASI algorithm was configured as presented in [31][3].

- **SASI-Gray-World:** this approach calculates gray world illuminant maps using $n = 1$, $p = 1$, and $\sigma = 3$ as the standard deviation of the Gaussian. The SASI descriptor is extracted from gray world IMs using the same configuration as SASI-IIC.

- **HOGedge-IIC:** we compute the HOGedge descriptor on the IIC-based illuminant map. For the HOGedge descriptor, it is necessary to adjust some parameters. We empirically determined the best parameters from training examples as: edge detection is performed on the Y channel of the YC_bC_r color space, with a Canny lower bound of 0 and an upper bound of 10. The square region for edge point filtering was set to

[3] We gratefully thank the authors for the source code.

32 × 32 pixels. Furthermore, we used 8-pixel cells without normalization in HOG, and 100 visual words for both the original and the tampered images (i. e., the dictionary consisted of 200 visual words).

- **HOGedge-Gray-World:** this configuration is similar to HOGedge-IIC. We computed gray world illuminant maps with $n = 1$, $p = 1$ and $\sigma = 3$. The (empirically determined) best performing parameters for HOGedge-Gray-World were the same as for HOGedge-IIC, with one exception: the size of the visual word dictionary was set to 75 visual words from each class (thus, the dictionary contained 150 visual words).
- **Metafusion:** We implemented a late fusion as explained in Section 3.5 using SASI-IIC, SASI-Gray-World, and HOGedge-IIC. We excluded HOGedge-Gray-World from the input methods, due to its relatively weak performance (see the evaluation below).

Figure 8 depicts a ROC curve of the performance of all methods using corner clicking marker. The area under the curve (AUC) is computed to obtain a single numerical measure for each result.

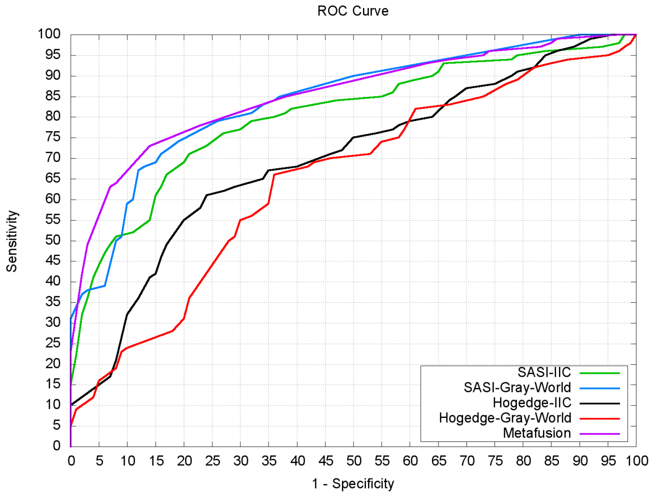


Figure 8: Comparison of different variants of the algorithm using semi-automatic (corner clicking) annotated faces.

We clearly see that Metafusion performs best, resulting in an AUC of 85.3%. In particular for high specificity (i. e., few false alarms), the method yields a much higher sensitivity compared to the other variants. Specifically, in a real forensic scenario, when an analyzed photograph is classified as composite using this variant of the method, it provides to an expert high confidence about the image authenticity. This kind of confidence is an important initial step when an expert needs to decide about image authenticity, decreasing the quantity of necessary future work.

The second best variant is SASI-Gray-World, with an AUC of 84.0%. In particular for a specificity below 80.0%, the sensitivity is comparable to Metafusion. SASI-IIC achieved an

AUC of 80.3%, followed by HOGedge-IIC with an AUC of 69.9% and HOGedge-Gray-World with an AUC of 64.7%.

4.3 Comparative Performance – Fully Automated versus Semi-Automatic Face Detection

We re-evaluated the best performing variant, Metafusion, with varying degrees of automation in the face detection and annotation.

- **Automatic Detection:** we used the PLS face detector [30] to detect faces in the images. In our experiments, the PLS face detector successfully located all present faces in only 65% of our images. We then performed a 3-fold cross validation on this 65% of the images. For training the classifier, we used the manually annotated bounding boxes. In the test images, we used the bounding boxes found by the automated detector.

- **Semi-Automatic Detection 1 (Eye Clicking):** an expert does not necessarily have to mark a bounding box. In this variant, the expert clicks in the eye positions. The Euclidean distance between the eyes is the used to construct a bounding box for the face area. For classifier training and testing we use the same setup and images as in the automatic detection.

- **Semi-Automatic Detection 2 (Corner Clicking):** in this variant, the expert has to click on the upper left and lower right corners of a face. These positions delimit the bounding box dimensions, making this the most accurate marker type. Once again we used the same training and testing setup as in the Automatic and Semi-Automatic Detection 1.

Figure 9 shows the results of this experiment. The semi-automatic detection using corner clicking resulted in an AUC of 78.0%, while the semi-automatic using eye clicking and the fully-automatic approaches yielded an AUC of 63.5% and AUC of 63.0%, respectively. Thus, as it can also be seen in Figures 10(a), 10(b) and 10(c), proper face location is important for improved performance. Although automatic face detection algorithms have improved over the years, we find user-selected faces more reliable for a forensic setup. Note, however, that the selection of faces under similar illumination conditions is a minor interaction, that requires no particular knowledge in image processing or image forensics.

5 Conclusions and Future Work

In this work, we presented a new method for detecting forged images of people using the color of the illuminant. We estimate the illuminant color with a statistical gray edge method and a physics-based method using the inverse intensity-chromaticity color space. We interpret these illuminant maps as texture maps and also extract edge information from them. To describe the edge information, we propose a new algorithm based on edge-points and the HOG descriptor, called HOGedge. We combine these complementary cues using machine

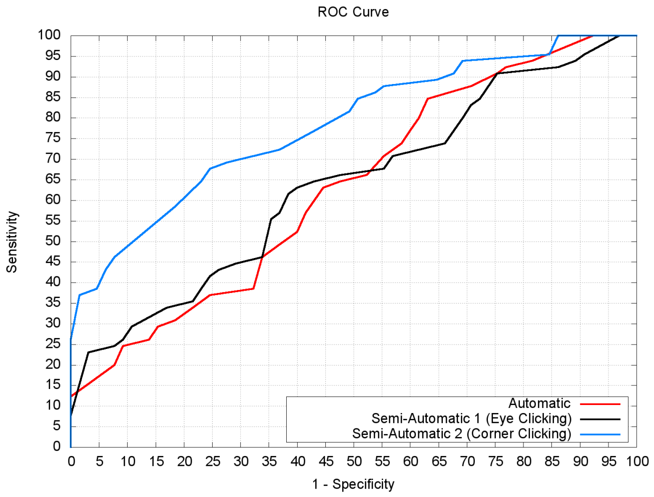
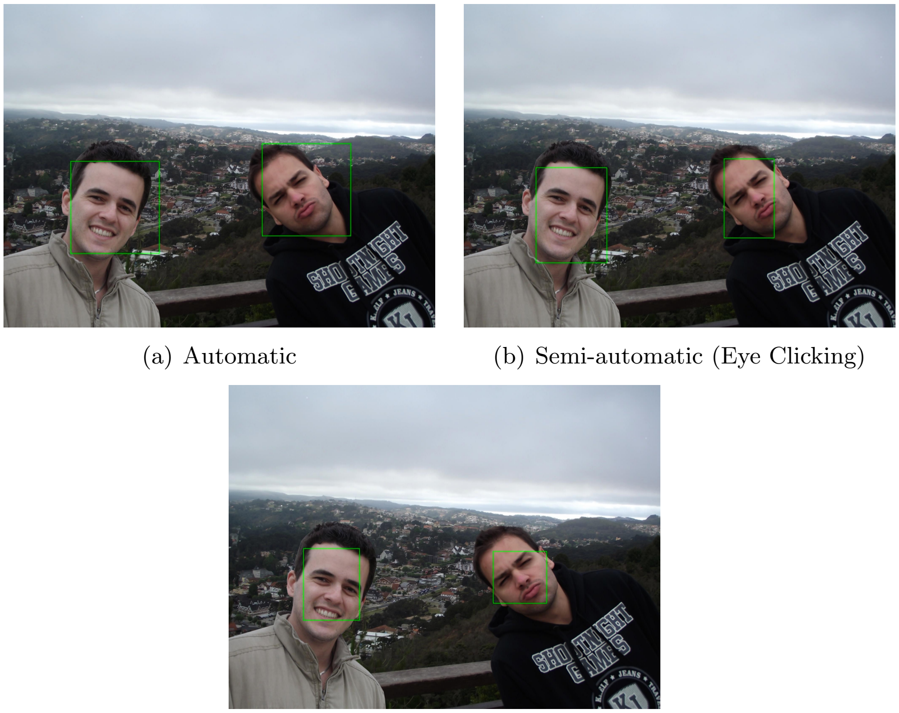


Figure 9: Experiments showing the differences for automatic and semi-automatic face detection.



(a) Automatic

(b) Semi-automatic (Eye Clicking)

(c) Semi-automatic (Corner Clicking)

Figure 10: Different types of face location. Automatic and semi-automatic locations select a considerable background part, whereas manual location is restricted to face region.

learning late fusion. Our results are encouraging, yielding an AUC of over 85% correct classification.

Although the proposed method is tailored to detect splicing on images containing faces, there is no principal hindrance in applying it to other, problem-specific materials in the scene.

The proposed method requires only a minimum of human interaction and provides a crisp statement on the authenticity of the image. Additionally, it is an important leap ahead to exploit color as a forensic cue. Prior color-based work either assumes complex user interaction or imposes very limiting assumptions.

Although promising as forensic cues, methods that operate on illuminant color are inherently prone to estimation errors. Thus, we expect that further improvements can be achieved when more advanced illuminant color estimators become available. For instance, while creating this article, Bianco and Schettini [39] proposed a machine-learning based illuminant estimator particularly for faces. An incorporation of this method is subject of future work.

Reasonably effective skin detection methods have been presented in the computer vision literature in the past years. Incorporating these cues can expand the scenario where our method could be applied to. Such an improvement could be employed, for instance, in detecting pornography compositions which according to forensic practitioners have become more and more common nowadays.

References

[1] A. Rocha, W. Scheirer, T. E. Boult, S. Goldenstein, Vision of the Unseen: Current Trends and Challenges in Digital Image and Video Forensics, ACM Computing Surveys 43 (2011) 1–42.

[2] H. Farid, A Survey of Image Forgery Detection, IEEE Signal Processing Magazine 2 (26) (2009) 16–25.

[3] H. Farid, A 3-D Lighting and Shadow Analysis of the JFK Zapruder Film (Frame 317), Tech. Rep. TR2010-677, Dartmouth College (2010).

[4] C. Riess, E. Angelopoulou, Scene Illumination as an Indicator of Image Manipulation, in: Information Hiding, Vol. 6387, 2010, pp. 66–80.

[5] H. Farid, M. J. Bravo, Image Forensic Analyses that Elude the Human Visual System, in: Symposium on Electronic Imaging (SPIE), 2010, pp. 1–10.

[6] Y. Ostrovsky, P. Cavanagh, P. Sinha, Perceiving Illumination Inconsistencies in Scenes, Perception 34 (11) (2005) 1301–1314.

[7] M. Johnson, H. Farid, Exposing digital forgeries by detecting inconsistencies in lighting, in: ACM Workshop on Multimedia and Security, ACM, New York, NY, USA, 2005, pp. 1–10.

[8] M. Johnson, H. Farid, Exposing Digital Forgeries in Complex Lighting Environments, IEEE Transactions on Information Forensics and Security (TIFS) 3 (2) (2007) 450–461.

[9] M. Johnson, H. Farid, Exposing Digital Forgeries through Specular Highlights on the Eye, in: International Workshop on Information Hiding, 2007, pp. 311–325.

[10] E. Kee, H. Farid, Exposing Digital Forgeries from 3-D Lighting Environments, in: IEEE International Workshop on Information Forensics and Security (WIFS), 2010, pp. 1–6.

[11] J. F. O'Brien, H. Farid, Exposing Photo Manipulation with Inconsistent Reflections, ACM Transactions on Graphics 31 (1) (2012) 1–11.

[12] S. Gholap, P. K. Bora, Illuminant Colour Based Image Forensics, in: IEEE Region 10 Conference, 2008, pp. 1–5.

[13] P. Saboia, T. Carvalho, A. Rocha, Eye Specular Highlights Telltales for Digital Forensics: A Machine Learning Approach, in: IEEE International Conference on Image Processing (ICIP), 2011, pp. 1937–1940.

[14] C. Riess, E. Angelopoulou, Physics-Based Illuminant Color Estimation as an Image Semantics Clue, in: IEEE International Conference on Image Processing, 2009, pp. 689–692.

[15] X. Wu, Z. Fang, Image Splicing Detection Using Illuminant Color Inconsistency, in: IEEE International Conference on Multimedia Information Networking and Security, 2011, pp. 600–603.

[16] J. van de Weijer, T. Gevers, A. Gijsenij, Edge-Based Color Constancy, IEEE Transactions on Image Processing (TIP) 16 (9) (2007) 2207–2214.

[17] K. Barnard, V. Cardei, B. Funt, A Comparison of Computational Color Constancy Algorithms – Part I: Methodology and Experiments With Synthesized Data, IEEE Transactions on Image Processing (TIP) 11 (9) (2002) 972–983.

[18] K. Barnard, L. Martin, A. Coath, B. Funt, A Comparison of Computational Color Constancy Algorithms – Part II: Experiments With Image Data, IEEE Transactions on Image Processing (TIP) 11 (9) (2002) 985–996.

[19] A. Gijsenij, T. Gevers, J. van de Weijer, Computational Color Constancy: Survey and Experiments, IEEE Transactions on Image Processing (TIP) 20 (9) (2011) 2475–2489.

[20] M. Bleier, C. Riess, S. Beigpour, E. Eibenberger, E. Angelopoulou, T. Tröger, A. Kaup, Color Constancy and Non-Uniform Illumination: Can Existing Algorithms Work?, in: IEEE Color and Photometry in Computer Vision Workshop, 2011, pp. 774–781.

[21] A. Gijsenij, R. Lu, T. Gevers, Color Constancy for Multiple Light Sources, IEEE Transactions on Image Processing 21 (2) (2012) 697–707.

[22] R. Kawakami, K. Ikeuchi, R. T. Tan, Consistent Surface Color for Texturing Large Objects in Outdoor Scenes, in: IEEE International Conference on Computer Vision, 2005, pp. 1200–1207.

[23] M. Ebner, Color Constancy using Local Color Shifts, in: European Conference in Computer Vision, 2004, pp. 276–287.

[24] P. F. Felzenszwalb, D. P. Huttenlocher, Efficient Graph-Based Image Segmentation, International Journal of Computer Vision (IJCV) 59 (2) (2004) 167–181.

[25] G. Buchsbaum, A Spatial Processor Model for Color Perception, Journal of the Franklin Institute 310 (1) (1980) 1–26.

[26] A. Gijsenij, T. Gevers, J. van de Weijer, Improving Color Constancy by Photometric Edge Weighting, IEEE Pattern Analysis and Machine Intelligence (PAMI) 34 (5) (2012) 918–929.

[27] R. Tan, K. Nishino, K. Ikeuchi, Color Constancy Through Inverse-Intensity Chromaticity Space, Journal of the Optical Society of America A 21 (2004) 321–334.

[28] C. Riess, E. Eibenberger, E. Angelopoulou, Illuminant Color Estimation for Real-World Mixed-Illuminant Scenes, in: IEEE Color and Photometry in Computer Vision Workshop, 2011.

[29] T. Igarashi, K. Nishino, S. K. Nayar, The Appearance of Human Skin: A Survey, Foundations and Trends in Computer Graphics and Vision 3 (1) (2007) 1–95.

[30] W. R. Schwartz, A. Kembhavi, D. Harwood, L. S. Davis, Human Detection Using Partial Least Squares Analysis, in: IEEE International Conference on Computer Vision (ICCV), 2009, pp. 24–31.

[31] O. A. B. Penatti, E. Valle, R. S. Torres, Comparative Study of Global Color and Texture Descriptors for Web Image Retrieval, Journal of Visual Communication and Image Representation 23 (2) (2012) 359–380.

[32] A. Çarkacioglu, F. T. Yarman-Vural, SASI: A Generic Texture Descriptor for Image Retrieval, Pattern Recognition 36 (11) (2003) 2615–2633.

[33] J. Canny, A Computational Approach to Edge Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 8 (6) (1986) 679–698.

[34] N. Dalal, B. Triggs, Histograms of Oriented Gradients for Human Detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 886–893.

[35] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, Visual Categorization With Bags of Keypoints, in: Workshop on Statistical Learning in Computer Vision, 2004, pp. 1–8.

[36] J. Winn, A. Criminisi, T. Minka, Object Categorization by Learned Universal Visual Dictionary, in: IEEE International Conference on Computer Vision (ICCV), 2005, pp. 1800–1807.

[37] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[38] O. Ludwig, D. Delgado, V. Goncalves, U. Nunes, Trainable Classifier-Fusion Schemes: An Application to Pedestrian Detection, in: IEEE International Conference on Intelligent Transportation Systems, 2009, pp. 1–6.

[39] S. Bianco, R. Schettini, Color Constancy using Faces, in: Proc. IEEE Computer Vision and Pattern Recognition, Providence, RI, USA, 2012.