ISSN 2229-5518

Detecting Digital Image Forgeries using Color Constancy

Jiji George

Abstract . There is a gradual increase in the number of composite pictures containing people. Due to the existence of such compositions, the trust in photographs is reduced. Due to the invention of powerful digital image editing tools, it has been so easy to manipulate images. Approaches that consider the illumination inconsistencies in digital images are of particular interest because a perfect illumination adjustment in a digital composite is very difficult to obtain. The proposed method is build upon the ideas by local illuminant color estimation of scenes, edge-based and texture-based color constancy analysis. Here, an illumination information provided by statistics based color constancy method is used. The method requires minimal user interaction for tampering decisions. For this, first the illuminant color is estimated using a statistical gray edge method, and treat this illuminant map as texture maps, then extract information using HOGedge algorithm and gabor texture features. These informations are then provided to a machine learning approach for automatic decision making. The classification is based on the well-known SVM classifier.

Index Terms— Image forensics, Digital tampering, Color constancy, Illuminant color, Machine learning, Spliced image detection, Texture and edge descriptors.

----

1. INTRODUCTION

The past few years have seen a considerable rise in the availability and sophistication of digital imaging technology (cameras, scanners, softwares) and their use in manipulating digital images. Everyday, millions of digital documents are produced by a variety of devices and distributed by newspapers, magazines, websites and television. Digital images are everywhere from our cell phones to the pages of our online news sites. In all these information channels, images are a powerful tool for communication. Investigators from a diverse set of fields require the best possible tools to tackle the challenges presented by the malicious use of today's digital image processing techniques. For decades, photographs have been used as evidence in courts and are often used for documenting space time events. Unfortunately, it is not difficult to use computer graphics and image processing techniques to manipulate images. Image composition (or splicing) is one of the most common image manipulation technique. The creation of digital forgery involves combining objects or people from different images. One such example is shown in Figure 1, in which the girl on the left is inserted.

• Jiji George, currently pursuing M.Tech Degree in Computer Science and Information technology from FISAT, Angamaly, Kerala, India

• E-mail: jijigeorge14@gmail.com

It has long been said that an image worth a thousand words. Recently, a study conducted by Italian Psychologists states that doctored photographs of past public events affect memory of those events. Their results indicate that doctored photographs of past public events can influence memory, attitudes and behavioral intentions.

Digital Image and Video Forensics research aims at analyzing the underlying facts about an image or video. Its main objectives comprise: tampering detection, hidden data detection or recovery and source identification with no prior measurement or registration of the image. Several techniques have been developed to detect various forms of digital tampering. When assessing the authenticity of an image, forensic investigators use all available sources of tampering evidence.

Figure 1: Example of a spliced image involving people

2. RELATED WORKS

ISSN 2229-5518

Alin C. Popescu et al.[1] proposed Color Filter Array Interpolation in images for detecting image manipulation. Each color image will have a Red, Green and Blue channels. Only a single color sample is recorded at each pixel location and the other two color samples must be estimated from the neighboring samples to obtain a three channel color image. The estimation of the missing color samples is known as CFA interpolation or demosaicking. The interpolation introduces specific correlations which will be altered when an image is manipulated. In this method, the correlation introduced using the color filter array interpolation are quantified and examine whether there is a change in correlation is occured.

Micah et al.[2] described a technique for exposing image manipulations on the basis of detecting inconsistencies in lighting. The main idea behind this technique is that, when an image forgery from multiple images are created, the process of matching lighting conditions is diffcult. This is because complex lighting environments produces different lighting gradients and shades in the image.

Hany Farid[3] proposed a method based on the difference in JPEG compression quality for detecting image manipulation. Different images are of different JPEG compression quality. This technique is based on the JPEG artifacts and has proven its efficiency in detecting tampering in low-quality images. In this method, the difference in pixel values are considered. When an image is forged, some variations will be occured inside and outside the tampered region. This difference is used as an evidence for image manipulation.

Xunyu et al.[4] described a region duplication detection method that is robust to distortions of the duplicated regions. This method starts by estimating the transform between matched scale invariant feature transform (SIFT) keypoints, which are insensitive to geometrical and illumination distortions, and then finds all pixels within the duplicated regions after discounting the estimated transforms.

Pravin Kakar et al.[5] proposed a novel technique based on transform-invariant features. It is based on the MPEG-7 image signature tools, which form a part of the MPEG-7 standard. This set of tools was designed for robust and fast image and video retrieval.

It is very difficult to properly size a foreign object, when it is inserted into an image especially when there is no reference object in the same distance. To detect this type of image forgery, a perspective-constraint based method is proposed by Heng Yao et al.[6], with which the height ratio of two objects in an image can be determined without any knowledge of the camera parameters.

Pravin Kakar et al.[7] proposed a novel method for detecting splicing in images, using discrepancies in motion. It uses the motion blur estimation through image gradients for detecting inconsistencies between the spliced region and the rest of the image. One of the possible causes of motion blur is the slow speed of the camera shutter relative to the object being imaged. In many images, camera shake is found to be the culprit for the presence of motion blur. Many images containing motion blur do exist and so, it is useful to utilize the inconsistencies in motion blur in order to detect image tampering.

A method is described by Micah et al.[8] that detects image forgery through specular highlights on the eye. The position of a specular highlight is determined by the relative positions of the light source, the reflective surface and the viewer (or camera). The light direction can be estimated from the surface normal and view direction at a specular highlight. The light direction is specified with respect to the eye.

Qiguang Liu et al.[10] proposed a framework for detecting tampered digital images based on photometric consistency of illumination in shadows. This method formulate color characteristics of shadows measured by the shadow matte value. The shadow boundaries and the penumbra shadow region(the less dark area) in an image are first extracted. Then a simple and efficient method is used to estimate shadow matte values of shadows.

3. FORGERY DETECTION

This project make an important step towards minimizing user interaction for an illuminant-based tampering decision making. This work proposes a new semiautomatic method that is also significantly more reliable than earlier approaches. Iluminant estimators are used to extract the texture based features and edge based features. These features are then provided to a machine learning technique for automatic decision making. Here exploit the fact that local illuminant estimates are most discriminative when comparing objects of the same (or similar) material. Thus, the method focuses on the automated comparison of human skin, and more specifically faces, to classify the illumination on a pair of faces as either consistent or inconsistent. The overview of the proposed method is shown in Figure.2.The method consists of five main components:

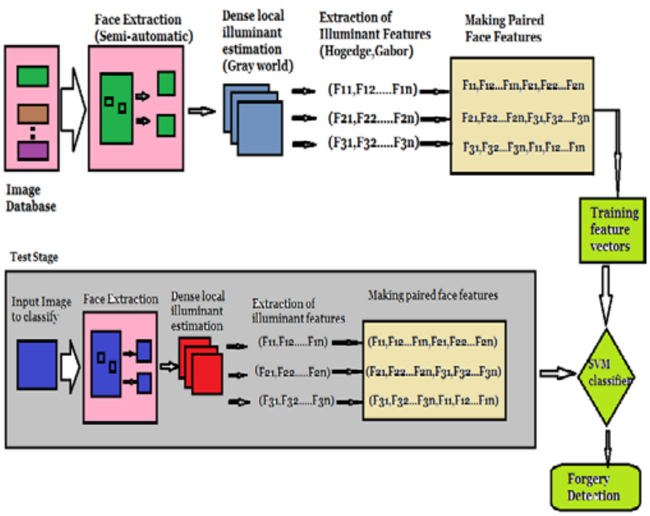

Figure 2: Overview of the proposed method

- Face Extraction: This is the only step that may require human interaction. An operator sets a bounding box around each face in the image that should be investigated.

- Dense local Illuminant Estimation(IE): The input image is segmented into homogeneous regions. For each illuminant estimator, a new image is created where each region is colored with the extracted illuminant color. This resulting intermediate representation is called illuminant map (IM).

- Computation of Illuminant Features: For all face regions, texture-based and gradient-based features are computed. Each one of them encodes complementary information for classification.

- Paired Face Features: The goal of this work is to determine whether a pair of faces in an image is consistently illuminated. So, for an image with faces, joint feature vectors are constructed, consisting of all possible pairs of faces.

- Classification: A machine learning approach is used to automatically classify the feature vectors. An image is considered as a forgery if atleast one pair of faces in the image is classified as inconsistently illuminated.

Face Extraction

Face is extracted from the input image. This is an important part of many biometric, security, and surveillance systems, as well as image and video indexing systems. The method detect digital image forgeries by analyzing the illumination inconsistencies in facial regions. Therefore the first step is to extract face regions from the image. Automated algorithms can be used for obtaining the bounding boxes,e.g.,the one by Schwartz et al.[10]. However, here prefer a human operator for this task for two main reasons:

- This minimizes false detections or missed faces;
- Scene context is important when judging the lighting situation.

Dense Local Illuminant Estimation

Dense local illuminant estimation is the process of creating the illuminant map(IM) of the extracted face regions of the input image. For creating the illuminant map, the generalized gray world approach is used. To compute a dense set of localized illuminant color estimates, the input image (face region) is segmented into regions of approximately constant chromaticity(superpixels), using the algorithm by Felzenszwalb and Huttenlocher [11]. The illuminant map creation is described as follows. The process of creating the illuminant map consist of two steps: Dividing the input image into super pixels and recoloring the superpixels.

For getting the super pixels, the image is divided into a number of segments. The RGB image is converted into grayscale by eliminating the hue and saturation information while retaining the luminance. This grayscale image is segmented using a graph based segmentation algorithm. For this method, threshold value(the size set for the segmentation area), the minimum size of the segmentation component and the number of nearest neighborhood of a pixel are considered. The K-nearest neighbor method is used. It create and return a matrix of the knn of each pixel and the corresponding distance of knng. The segmentation algorithm returns the labeled image, in which different areas are labeled by different numbers. Thus we get the different segments, each of which have approximately constant chromaticity, called the super pixels.

To get the illuminant map, the superpixels have to be recolored using the extracted illuminant color. Let

$$f(x) = (f_R(x), f_G(x), f_B(x))^T$$

denote the observed RGB color of a pixel at location x. The values and id of the Red, Green and Blue components are calculated and are used to estimate the light source of the input image. Per superpixel, the color of the illuminant is estimated. The color of the illuminant e is estimated as:

ISSN 2229-5518

$$\mathrm{k}e^{n,p,\sigma}=\left(\int\left(\frac{\partial^n f^{\sigma}(x)}{\partial x^n}\right)dx\right)^{1/p}$$

Here, the integral is computed over all pixels in the image, where x denotes a particular position (pixel coordinate). Furthermore, k denotes a scaling factor, ∂ the differential operator, and f^{σ}(x) the observed intensities at position x, smoothed with a Gaussian kernel σ.

The generalized grayworld approach used here is an extension of the classical grayworld assumption by Buchsbaum[12]. The classical grayworld approach states that the average color of a scene is gray. Thus a deviation of the image intenseties from the expected gray color is due to the illuminant. But this assumption is considered overly simpli_ed and hence it has inspired further improvement. The extension of this idea is the generalized grayworld approach by van de Weijer et al[13].

The generalized grayworld approach incorporated three parameters to extend the classical grayworld method:

- Derivative order n : The assumption that the average of the illuminants is achromatic and can be extended to the absolute value of the sum of the derivatives of the image.
- 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.
- Gaussian smoothing σ: To reduce image noise, one can smooth the image prior to processing with a Gaussian kernel of standard deviation.

Computation of Illuminant Features

After creating the illuminant map, illuminant features are computed. For the feature computation, two algorithms are used: The HOGedge algorithm and Gabor method.

1) The HOGedge algorithm

From the illuminant map created using the grayworld approach, the edge features are extracted using the HOGedge method. The HOGedge features are computed around the edge points of the face regions in the illuminant map. The main idea behind this method is that, when an image is spliced, the statistics of the edges is likely to differ from original images. The HOGedge method is based on the well-known HOG-descriptor, and computes visual dictionaries of gradient intensities in edge points. Approximately equally distributed candidate points on the edges of illuminant maps are extracted first. At these points, HOG descriptors are computed. These descriptors are summarized in a visual words dictionary.

The process of extraction of the HOGedge features from the illuminant map can be divided into 3 steps:

1.Extraction of Edge Points: Given a face region from an illuminant map, the edge points are extracted using the Canny edge detector [14]. The Canny edge detector detect a combination of the segment borders with similar incident light in the image. This yields a large number of spatially close edge points. To reduce the number of points, the Canny output is filtered using the following rule: Starting from a seed point, 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 for the next iteration. By iterating this process over the entire image, we reduce the number of points but still ensure that every face has a comparable density of points.

2. Point Description: Next compute Histograms of Oriented Gradients (HOG)[15] to describe the distribution of the selected edge points. HOG is based on normalized local histograms of image gradient orientations in a dense grid. The HOG descriptor is constructed around each of the edge points. The neighborhood of such an edge point is called a cell. Each cell provides a local 1-D histogram of quantized gradient directions using all cell pixels. To construct the feature vector, the histograms of all cells within a spatially larger region are combined and contrast Normalized. Then use the HOG output as a feature vector for the subsequent steps.

3. Visual Vocabulary Creation: The number of extracted HOG vectors varies depending on the size and structure of the face under examination. Then visual dictionaries [16] are used to obtain feature vectors of fixed length. Visual dictionaries constitute a robust representation, where each face is treated as a set of region descriptors. The spatial location of each region is discarded.

The visual dictionary can be created by dividing the training data into feature vectors from original and doctored images. Each group is clustered using the k-means algorithm [17]. Then, a visual dictionary with visual words is constructed, where each word is represented by a cluster center. Thus, the visual dictionary summarizes the most representative feature vectors of the training set. The pseudocode for the creation of visual dictionary is given as follows:

Algorithm 1 HOGedge—Visual dictionary creation

Require: V_{TR} (training database examples)n(the number of visual words per class)
Ensure: V_D (visual dictionary containing $2n$ visual words)

```
V_D ← ∅;
V_NF ← ∅;
V_DF ← ∅;
for each face IM i ∈ V_TR do
    V_EP ← edge points extracted from i ;
    for each point j ∈ V_EP do
        FV ← apply HOG in image i at position j;
        if i is a doctored face then
            V_DF ←{V_DF ∪ FV};
        else
            V_NF ←{V_NF ∪ FV};
        end if
    end for
end for
Cluster V_DF using n centers;
Cluster V_NF using n centers; V_D ←{centers of V_DF ∪ centers of V_NF};
return V_D;
```

Algorithm 2 HOGedge—Face characterization

Require: V_D (visual dictionary precomputed with $2n$ visual words) IM (illuminant map from a face)
Ensure: HFV (HOGedge feature vector)

```
HFV ← 2n-dimensional vector, initialized to all zeros;
V_FV ← ∅;
V_EP ← edge points extracted from IM;
for each point i ∈ V_EP do
    FV ← apply HOG in image IM at position j;
    V_FV ←{V_FV ∪ FV};
end for
for each feature vector i ∈ V_FV do
    lower_distance ← +∞;
    position ← −1;
    for each visual word j ∈ V_D do
        distance ← Euclidean distance between i and j;
        if distance < lower_distance then
            lower_distance ← distance;
            position ← position of j in V_D;
        end if
    end for
    HFV [position] ← HFV [position] + 1;
end for
return HFV;
```

For the visual vocabulary creation, the training database VTR is required which contains all the images for training. n is the number of visual words per class. VD is the visual vocabulary created using this algorithm, VNF is the feature set of normal images and VDF is the feature set of doctored images. For each face of the images in the training database, the extracted edge points are stored in VEP. Then apply the HOG algorithm at each edge point and store the resulting HOGedge features in FV. If the image that contains this face is a doctored one, then add this feature vector FV with the feature set of doctored images, VDF. If the image that contains this face is an original one, then add this feature vector FV with the feature set of normal images, VNF. After applying these steps in all the edge points of every faces in the database, cluster VDF and VNF using n clusters. Each cluster is represented using the cluster centers. VD is obtained by taking the union of centers of VDF and VNF.

Quantization Using the Precomputed Visual Dictionary:
For evaluation, the HOG feature vectors are mapped to the visual dictionary. Each feature vector in an image is represented by the closest word in the dictionary. (with respect to the Euclidean distance). A histogram of word counts represents the distribution of HOG feature vectors in a face. The pseudocode for face characterization using the HOGedge algorithm is given as follows:

For the face characterization using the HOGedge method, the visual dictionary VD and the Illuminant map IM from all the faces in the image to be tested are required. HFV is a 2n-dimensional HOGedge feature vector created using this algorithm. VEP is the edge points extracted from the illuminant map IM of the test image. For each such edge points, HOG is applied and the feature vector FV is calculated. The feature vectors for all the edge points in the test image are stored in VFV. For each feature vector in VFV , the lower distance is set as 1 and the position is set as 1. Then the Euclidean distance between this feature vector and the visual words in VD is calculated and select the visual word that has the least distance with the feature vector. So the feature vector is represented using the closest word in the dictionary and this word is returned as HFV.

1) The Gabor method

Features constructed from responses of Gabor filters, Gabor features[18], have been particularly successful in many computer vision and image processing applications. Gabor features extract local pieces of information which are then combined to recognize an object or region of interest. The core of Gabor filter based feature extraction is the 2D Gabor filter function, which is given in the following equations.

$x' = x\cos\theta + y\sin\theta$

y′ = -xsinθ + ycosθ

$\varphi(x,y)$ =(f^2/ ($\pi*\gamma*\eta$))*exp(-((f^2/γ^2)* $x^{'2}$)+($f^2/\eta^2)*$ $y^{'2}$))*exp(j*2*π*f*$x^{'}$)

In the spatial domain, the Gabor filter is a complex plane wave (a 2D Fourier basis function) multiplied by an origin centered Gaussian. f is the central frequency of the filter, θ is the rotation angle, γ is the sharpness (bandwidth) along the Gaussian major axis, and η is the sharpness along the minor axis (perpendicular to the wave).

Face Pair

To compare two faces, the same descriptors for each of the two faces are combined. The idea is that, feature concatenation from two faces is different when one of the faces is an original and one is spliced. For an image containing n_f faces (n_f ≥2), the number of face pairs is (nf (nf-1))/2.

Classification

The illumination for each pair of faces in an image is classified as either consistent or inconsistent. Assuming all selected faces are illuminated by the same light source, an image is tagged as manipulated if atleast one pair is classified as inconsistent. Individual feature vectors, i.e., HOGedge features and gabor features are classified using a support vector machine (SVM) classifier with a radial basis function (RBF) kernel. The HOGedge algorithm and the Gabor method individually analyses the input image by considering the illuminant map created for the respective faces and give a crisp statement about the authenticity of that image. The output can be either as 'Original image' or as 'Forged image'. If one of the method predicted it as 'forged image' then the image can considered as forged because both HOGedge and Gabor method analyses different types of features for the analysis and classification.

4. EXPERIMENTS

In this project, the processing is done on grayscale images. In case of RGB, the image is converted to grayscale for processing.

The training is performed using a dataset containing 100 images. Out of this set of images, 50 are original i.e. images that have no adjustments and the remaining 50 are forged. Some of these images are captured by ourselves while some of them were collected from the internet. During training the user have to specify the number of images in the database that is going to be used for training. For this, the first half of the total images in the database should be normal images and the remaining half should be forged images. Then all the images in the database are processed in the order. Face recognition is the process of identifying one or more people in images or videos. This is an important part of many biometric, security, and surveillance systems, as well as image and video indexing systems. The method detect digital image forgeries by analyzing the illumination inconsistencies in facial regions. Therefore the first step in processing is to extract face regions from the input image. For each input image the number of faces present in that image is specified. Then the faces are extracted from the image. After face extraction , the illuminant map is created for each face in the image. From the illuminant map, the edges of the face is detected using canny edge detector and these edge points are considered for extracting the HOGedge features. The gabor texture features are also extracted. During the experiment, 20 images were tested that contain both original and forged images. Among this test set, 17 images were correctly classified. So,the proposed system exhibited an accuracy of 85%.

5. CONCLUSION

In this paper, a new method is proposed for detecting forged images of people using color constancy. The proposed method includes face extraction from the input image and an illuminant map is created for all faces. Then for all face regions, edge-based and texture-based features are computed. Each one of them encodes complementary information for classification. Then paired face features are considered to determine whether there is a manipulation is occured. A machine learning approach is used to automatically classify the feature vectors. An image is considered as a forgery if at least one pair of faces in the image is classified as inconsistent. Here estimate the illuminant color using gray world method and interpret this illuminant map as texture map and also extract edge information from them. To describe the edge information, a new algorithm based on edge-points, called the HOGedge is used. The texture features are extracted using the well known gabor features. 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 step to exploit color as a forensic cue. Prior color based work either assumes complex user interaction or imposes very limiting assumptions. Although the proposed method is intended to detect splicing on images containing faces.

6. REFERENCES

[1]. Alin C. Popescu and Hany Farid, "Exposing Digital Forgeries in Color Filter Array Interpolated Images",IEEE

TRANSACTIONS ON SIGNAL PROCESSING, VOL. 53, NO. 10, OCTOBER 2005.

[2] Micah K. Johnson, and Hany Farid, "Exposing Digital Forgeries in Complex Lighting Environments", IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 2, NO. 3, SEPTEMBER 2007.

[3] Hany Farid,"Exposing Digital Forgeries From JPEG Ghosts",IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 4, NO. 1, MARCH 2009.

[4] Xunyu Pan and Siwei Lyu, "Region Duplication Detection Using Image Feature Matching",IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 5, NO. 4, DECEMBER 2010.

[5] Pravin Kakar, and N. Sudha, "Exposing Postprocessed CopyPaste Forgeries Through Transform-Invariant Features", IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 7, NO. 3, JUNE 2012.

[6] Heng Yao, Shuozhong Wang, Yan Zhao, and Xinpeng Zhang,"Detecting Image Forgery Using Perspective Constraints",IEEE SIGNAL PROCESSING LETTERS, VOL. 19, NO. 3, MARCH 2012.

[7] Pravin Kakar, N. Sudha and Wee Ser, "Exposing Digital Image Forgeries by Detecting Discrepancies in Motion Blur",IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 13, NO. 3, JUNE 2011.

[8] Micah K. Johnson and Hany Farid,"Exposing Digital Forgeries Through Specular Highlights on the Eye".

[9] Qiguang Liu, Xiaochun Cao, Chao Deng, and Xiaojie Guo,"Identifying Image Composites Through Shadow Matte Consistency",IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY,VOL. 6, NO. 3, SEPTEMBER 2011.

[10] W. R. Schwartz, A. Kembhavi, D. Harwood, and L. S. Davis, Human detection using partial least squares analysis, in Proc. IEEE International Conference on Computer Vision (ICCV), 2009, pp. 2431.

[11] P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient graph-based image segmentation,” *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, 2004.

[12] G. Buchsbaum, A spatial processor model for color perception, J. Franklin Inst., vol. 310, no. 1, pp. 126, Jul. 1980.

[13] J. van de Weijer, T. Gevers, and A. Gijsenij, Edge-based color constancy, IEEE Transaction on Image Processing, vol. 16, no. 9, pp. 22072214, Sep. 2007.

[14] J. Canny, A computational approach to edge detection, IEEE Transaction on Pattern Analalysis and Machine Intelligence., vol. 8, no. 6, pp. 679698, Jun. 1986.

[15] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2005, pp. 886–893.

[16] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, Visual categorization with bags of keypoints, in Proc. Workshop on Statistical Learning in Computer Vision, 2004, pp. 1-8.

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

[18] Joni-Kristian Kamarainen,"Gabor Features in Image Analysis".