COMPAQ

Statistical Color Models with Application to Skin Detection

Michael J. Jones *James M. Rehg*

Cambridge Research Laboratory

Technical Report Series

CRL 98/11

December 1998

Cambridge Research Laboratory

The Cambridge Research Laboratory was founded in 1987 to advance the state of the art in both core computing and human-computer interaction, and to use the knowledge so gained to support the Company's corporate objectives. We believe this is best accomplished through interconnected pursuits in technology creation, advanced systems engineering, and business development. We are actively investigating scalable computing; mobile computing; vision-based human and scene sensing; speech interaction; computer-animated synthetic persona; intelligent information appliances; and the capture, coding, storage, indexing, retrieval, decoding, and rendering of multimedia data. We recognize and embrace a technology creation model which is characterized by three major phases:

Freedom: The life blood of the Laboratory comes from the observations and imaginations of our research staff. It is here that challenging research problems are uncovered (through discussions with customers, through interactions with others in the Corporation, through other professional interactions, through reading, and the like) or that new ideas are born. For any such problem or idea, this phase culminates in the nucleation of a project team around a well articulated central research question and the outlining of a research plan.

Focus: Once a team is formed, we aggressively pursue the creation of new technology based on the plan. This may involve direct collaboration with other technical professionals inside and outside the Corporation. This phase culminates in the demonstrable creation of new technology which may take any of a number of forms - a journal article, a technical talk, a working prototype, a patent application, or some combination of these. The research team is typically augmented with other resident professionals—engineering and business development—who work as integral members of the core team to prepare preliminary plans for how best to leverage this new knowledge, either through internal transfer of technology or through other means.

Follow-through: We actively pursue taking the best technologies to the marketplace. For those opportunities which are not immediately transferred internally and where the team has identified a significant opportunity, the business development and engineering staff will lead early-stage commercial development, often in conjunction with members of the research staff. While the value to the Corporation of taking these new ideas to the market is clear, it also has a significant positive impact on our future research work by providing the means to understand intimately the problems and opportunities in the market and to more fully exercise our ideas and concepts in real-world settings.

Throughout this process, communicating our understanding is a critical part of what we do, and participating in the larger technical community—through the publication of refereed journal articles and the presentation of our ideas at conferences–is essential. Our technical report series supports and facilitates broad and early dissemination of our work. We welcome your feedback on its effectiveness.

Robert A. Iannucci, Ph.D.
Director

Statistical Color Models with Application to Skin Detection

Michael J. Jones James M. Rehg

December 1998

Abstract

The existence of large image datasets such as the set of photos on the world wide web make it possible to build powerful generic models for low-level image attributes like color using simple histogram learning techniques. We describe the construction of color models for skin and non-skin classes from a dataset of nearly 1 billion labelled pixels. These classes exhibit a surprising degree of separability which we exploit by building a skin pixel detector achieving a detection rate of 80% with 8.5% false positives. We compare the performance of histogram and mixture models in skin detection and find histogram models to be superior in accuracy and computational cost. Using aggregate features computed from the skin detector we build a surprisingly effective detector for naked people. Our results suggest that color can be a powerful cue for detecting people in unconstrained imagery. We believe this work is the most comprehensive and detailed exploration of skin color models to date.

Author email: {mjones,rehg}@crl.dec.com

CRL Technical reports are available on the CRL's web page at
http://www.crl.research.digital.com.

Compaq Computer Corporation
Cambridge Research Laboratory
One Kendall Square, Building 700
Cambridge, Massachusetts 02139 USA

1 Introduction

A central task in visual learning is the construction of statistical models of image appearance from pixel data. A solution consists of a representation of image appearance, a learning algorithm, and a source of training images. When the amount of available training data is small, sophisticated learning algorithms may be required to interpolate between samples. However, as a result of the world wide web and the proliferation of on-line image collections such as Corbis [2], the vision community today has access to image libraries of unprecendented size and richness.[1] These large data sets can support simple, computationally efficient learning algorithms. However, a data set such as web images constitutes a biased sample from the space of possible imagery. Thus, the process of building image models from web data must be accompanied by a process of visualizing these models and investigating the statistical characteristics of on-line image data sets.

Color is the simplest attribute of the set of pixels that make up an image. Recently a number of authors have addressed the problem of constructing "generic prior models" [26] of images using multi-scale statistical modeling techniques [8, 27, 18, 19, 3, 4]. In these approaches, texture models are constructed from the outputs of multi-scale spatial filters, such as wavelets or steerable pyramids. In most of this work, image models are built from a single example image, or a few examples in the case of [4]. Applications include texture synthesis and classification, as well as noise removal and image coding. A statistical color model can be viewed as the *0th* order version of these spatial models in which the neighborhood structure is limited to a single pixel. Color is the logical starting point for constructing generic models from large data sets, as it is computationally inexpensive and easy to visualize since the state space is three dimensional.

This report describes the construction of statistical color models from a data set of unprecedented size: Our model includes nearly 1 billion labeled training pixels obtained from random crawls of the world wide web. From this data we construct a generic color model as well as separate skin and non-skin models. We use visualization techniques to examine the shape of these distributions. We show empirically that the preponderance of skin pixels in web images introduces a systematic bias in the generic distribution of color. We learn both histogram and mixture densities from this data, and show that histogram models slightly outperform mixture models in this domain.

We use skin and non-skin color models to design a skin pixel classifier with an equal error rate of 88%. This is surprisingly good performance given the unconstrained nature of web images. Our visualization studies demonstrate the separation between skin and non-skin color distributions that make this performance possible. Using our skin classifier, we construct a system for detecting images containing naked people, based on simple aggregate properties of the classifier output. This system compares favorably to recent systems by Forsyth *et al.* [6] and Wang *et al.* [24]. This suggests that skin color can be a more powerful cue for detecting people in unconstrained imagery

[1] Informal estimates put the number of color photos currently available on the world wide web at 40 million [20].

than was previously suspected.

We believe this work is the most comprehensive and detailed exploration of skin color models to date. We plan to make available to the academic community the labeled dataset of 13,640 photos on which this report is based. See Appendix A for details.

Section 2 describes the construction and visualization of histogram color models. These models are applied to skin classification in Section 3, where they are also contrasted to mixture densities. Section 4 explores the application of the skin detector to image classification. We review previous work in Section 5 and discuss our conclusions and future plans in Section 6. The Appendix contains a number of facts about our dataset and classifier.

2 Histogram Color Models

There are two issues that must be addressed in building a color histogram model: the choice of color space and the size of the histogram, which is measured by the number of bins per color channel. The RGB color space is a natural representation for color images found on the web. While alternative color spaces such as YUV or HSV are available, RGB is the *de facto* standard for image representations such as JPEG.[2]

Web images fit naturally into a 24 bit color representation, since high quality color images require 24 bits and images with coarser color resolutions can be mapped into it. In contrast, the size of the histogram depends upon the task. Our starting point for color analysis is the direct construction of a histogram color model in 24 bit RGB color space. Such a model has a size of 256 bins per color channel, which corresponds to more than 16.7 million (256^3) bins, each mapped to a specific R,G,B color triple. In Section 3 we will show that skin classification requires a smaller histogram size for good generalization.

The dataset for the experiments described in this report were obtained by a large crawl of the web which produced about 3 million images (including icons and graphics). A smaller set of images was randomly sampled from this large set and cleared of all icons and graphics by hand. This produced a set of 18,696 photographs. This set was then manually separated into a set of 9731 images containing skin and 8965 images not containing any skin. This is a dataset of nearly 2 billion pixels, which is two orders of magnitude more data than the number of degrees of freedom in a histogram model of size 256. In Section 2.1 this dataset is used to build a general color model.

In Section 2.2 we used a subset of 13,640 photos to build specialized skin and non-skin color models. The regions of skin in 4675 skin images were segmented by hand as described in Appendix A. This set in conjunction with the 8965 non-skin images gives a total of nearly 1 billion labelled pixels. Details on how to obtain this dataset for academic research purposes can be found in Appendix A.

[2] The JPEG standard [22] does not specify a color space, but RGB is the most common representation for JPEG-encoded web images.

2.1 General Color Model

We first learn a general color model using a histogram of size 256 in RGB space. Each of the three histogram dimensions is divided into 256 bins, and each bin stores an integer counting the number of times that color value occurred in the entire database of images. The pixels in the 18,696 photograph dataset were used to populate the histogram. The histogram counts are converted into a discrete probability distribution $P(\cdot)$ in the usual manner:

$$P(rgb) = \frac{c[rgb]}{T_c}, \tag{1}$$

where $c[rgb]$ gives the count in the histogram bin associated with the RGB color triple rgb and T_c is the total count obtained by summing the counts in all of the bins.

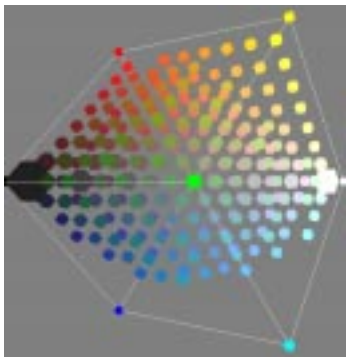
To visualize the probability distribution, we developed a software tool for viewing the histogram as a 3-D model in which each bin is rendered as a cube whose size is proportional to the number of counts it contains. The color of each cube corresponds to the smallest RGB triple which is mapped to that bin in the histogram. Figure 1 (a) shows a sample view of the histogram, produced by our tool. This rendering uses a perspective projection model with a viewing direction along the green-magenta axis which joins corners $(0, 255, 0)$ and $(255, 0, 255)$ in color space. The viewpoint was chosen to orient the gray line horizontally. The gray line is the projection of the gray axis which connects the black $(0, 0, 0)$ and white $(255, 255, 255)$ corners of the cube. The histogram in Figure 1 (a) is of size 8 and only shows bins with counts greater than $336,818$. Down-sampling and thresholding the full size model makes the global structure of the distribution more visible.

By examining the 3-D histogram from several angles its overall shape can be inferred. Another visualization of the model can be obtained by computing its marginal distribution along a viewing direction and plotting the resulting 2-D density function as a surface. Figure 1 (b) shows the marginal distribution that results from integrating the 3-D histogram along the same green-magenta axis used in Figure 1 (a). The positions of the black-red and black-green axes under projection are also shown. The density is concentrated along a ridge which follows the gray line from black to white. White has the highest likelihood, followed closely by black.

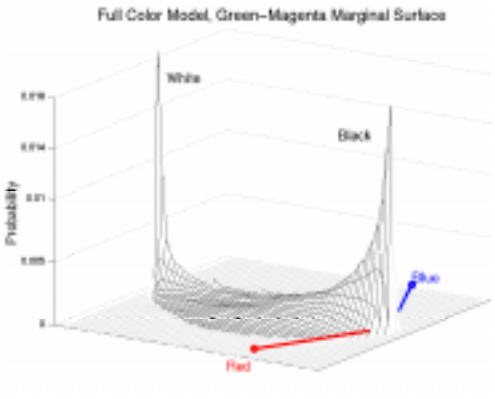
Additional information about the shape of the surface in Figure 1 (b) can obtained by plotting its equiprobability contours. These are shown in Figure 1 (c). They were obtained with the `contour` function in Matlab 5.0. It is useful to compare Figure 1 (c) with Figure 1 (a) as they are drawn from the same viewpoint. This plot reinforces the conclusion that the density is concentrated around the gray line and is more sharply peaked at white than black. An intriguing feature of this plot is the bias in the distribution towards red.

This bias is clearly visible in Figure 1 (d), which shows the contours produced by a different marginal density, obtained by integrating along the gray axis. The distribution shows a marked asymmetry with respect to the axis of projection that is oriented at approxmiately 30 degrees to the red line in the figure. In the next section, we will demonstrate empirically that this bias is due largely to the presence of skin in web images.

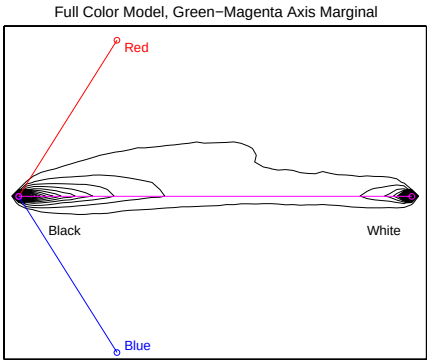
In summary, the generic color model built from web images has three properties:



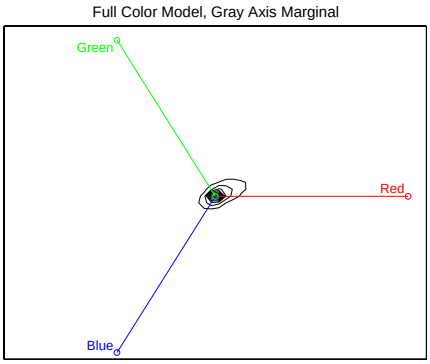
(a) 2-D rendering of 3-D histogram model viewed along the green-magenta axis.



(b) Surface plot of the marginal density formed by integrating along the viewing direction in (a).



(c) Equiprobability contours from the surface plot in (b).



(d) Contour plot for an integration of (a) along the gray axis.

Figure 1: Four visualizations of a full color RGB histogram model constructed from nearly 2 billion web image pixels.

1. Most colors fall on or near the gray line.

2. Black and white are by far the most frequent colors, with white occuring slightly more frequently.

3. There is a marked skew in the distribution toward the red corner of the color cube.

In gathering our dataset we made two additional interesting observations about images on the web. First, 77% of the possible 24 bit RGB colors are never encountered (i.e. the histogram is mostly empty). Second, about 52% of our web images have people in them. Table 1 contains a summary of facts about our dataset and color models.

2.2 Skin and Non-skin Color Models

A generic color model can be specialized to describe particular classes of objects if labels are available for the training pixels. We now address the construction of histogram color models for skin and non-skin pixel classes.

The color of skin in images depends primarily on the concentration of hemoglobin and melanin and on the conditions of illumination. It is well-known that the hue of skin is roughly invariant across different ethnic groups after the illuminant has been discounted. This is because differences in the concentration of pigments primarily affect the saturation of skin color, not the hue.

Unfortunately we do not know the illumination conditions in an arbitrary image[3] and so the variation in skin colors is much less constrained in practice. This is particularly true for web images captured under a wide variety of conditions. However, given a large collection of labeled training pixels we can still model the distribution of skin and non-skin colors in un-normalized color space.

We constructed skin and non-skin histogram models using our 13,640 photo dataset. The skin pixels in the 4675 images containing skin were labelled manually and placed into the skin histogram. The 8965 images that did not contain skin were placed into the non-skin histogram. Appendix A describes the labelling process in detail.

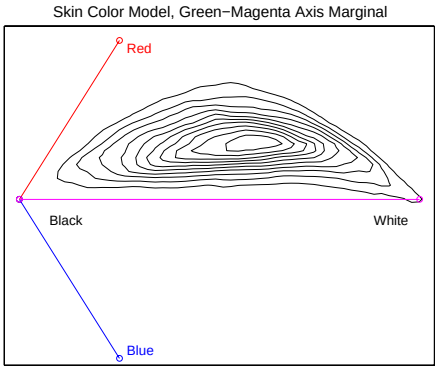
Given skin and non-skin histograms we can compute the probability that a given color value belongs to the skin and non-skin classes:

$$P(rgb|skin) = \frac{s[rgb]}{T_s}, \qquad P(rgb|\neg skin) = \frac{n[rgb]}{T_n} \tag{2}$$

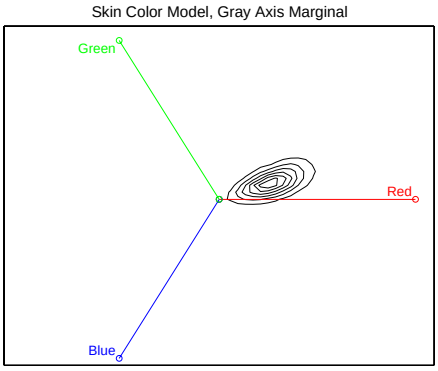
where $s[rgb]$ is the pixel count contained in bin rgb of the skin histogram, $n[rgb]$ is the equivalent count from the non-skin histogram, and T_s and T_n are the total counts contained in the skin and non-skin histograms, respectively.

The skin and non-skin color models can be examined using the same techniques we employed with the full color model. Contour plots for marginalizations of the skin and non-skin models are shown in Figure 2. The marginalizations are formed by integrating the distribution along two orthogonal viewing axes. These plots show that a significant

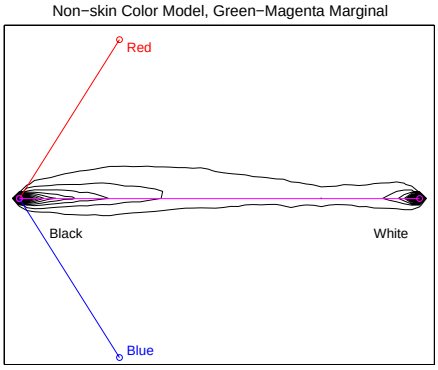
[3]The illuminant could be discounted, however, if a solution to the color constancy problem [23] were available.



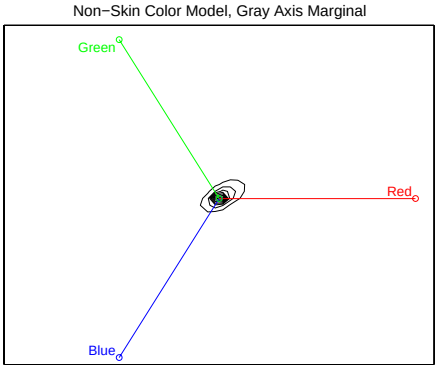
(a) Contour plot for skin model, marginalized along the green-magenta axis.



(b) Contour plot for skin model, marginalized along the gray axis.



(c) Contour plot for non-skin model, marginalized along the green-magenta axis.



(d) Contour plot for non-skin model, marginalized along the gray axis.

Figure 2: Contour plots for marginalizations of the skin and non-skin color models. The top row shows the skin model, the bottom row shows the non-skin model. The left column uses the viewing direction from Figure 1 (c) while the right column uses the view from Figure 1 (d).

	Total Counts	Total Occupied Bins	Percent Unoccupied
General Model	1,949,659,888	3,925,498	76.6
Skin Model	80,377,671	959,955	94.3
Non-skin Model	854,744,181	3,273,160	80.5
Overlapping skin/non-skin bins:			933,275
Skin pixels as a percentage of total pixels:			10 %
Total photos in labeled dataset:			13,640
Percentage of photos containing skin:			52 %

Table 1: Facts about photo image data set and the general, skin, and non-skin color models that were constructed from it.

degree of separation exists between the skin and non-skin models. The non-skin model, is concentrated along the gray axis, while the majority of the probability mass in the skin model lies off this axis. This separation between the two classes is the basis for the good performance of our skin classifier, which will be described in Section 3.

It is interesting to compare the non-skin color model illustrated in Figure 2 (c) and (d) with the full color model shown in Figure 1 (c) and (d). The only difference in the construction of these two models is the absence of skin pixels in the non-skin case. Note that the result of omitting skin pixels is a remarkable increase in the symmetry of the distribution around the gray axis. This observation suggests that although skin pixels constitute only about 10% of the total pixels in the dataset, they exhert a disproportionately large effect on the shape of the generic color distribution for web images, biasing it strongly in the red direction. We suspect that this effect results from the fact that the skin class occurs more frequently than other classes of object colors (52 % of our images contained skin).

2.3 Discussion

A number of statistics about the general, skin, and non-skin histogram color models are summarized in Table 1. Total counts gives the total number of pixels used to form each of the three models.[4] Note that the skin model was formed from more than 80.3 million hand labelled skin pixels! Total occupied bins refers to the number of bins in each model with nonzero counts. This is also expressed as the percentage of the bins in each model that were unoccupied. Overlapping bins gives the number of bins which are non-empty in both skin and non-skin histogram models.

We can make a few interesting observations about these statistics. First, 76.6% of the 16.7 million possible RGB values were not encountered in any of the training images. Second, of the 959,955 colors that occurred as skin, 933,275 (97.2%) also occurred as non-skin. This suggests that the skin detection problem could be difficult since there is significant overlap between the skin and non-skin models. However, overlap is only a significant problem if the counts in the shared bins are comparable

[4]The general model was constructed from 18,696 photos, while the skin and non-skin models were constructed from 13,640 photos. See Appendix A for details.

in the skin and non-skin cases. The plots in Figure 2 demonstrate that there is in fact reasonable separation between the skin and non-skin classes.

3 Skin Detection Using Color Models

Given skin and non-skin histogram models we can construct a skin pixel classifier. Such a classifier could be extremely useful in two contexts. First, for applications such as the detection and recognition of faces and figures, skin is a useful low-level cue that can be used to focus attention on the most relevant portions of an image. This approach is used in many systems, see [16, 6] for two examples. A second role for skin pixel detection is in image indexing and retrieval, where the presence of skin pixels in a photo is an attribute that could support queries or categorization. We give two examples of this application in Section 4.

The key step in skin pixel classification is the computation of $P(skin|rgb)$, which is given by Bayes rule:

$$P(skin|rgb) = \frac{P(rgb|skin)P(skin)}{P(rgb|skin)P(skin) + P(rgb|\neg skin)P(\neg skin)}. \quad (3)$$

A particular RGB value is labelled skin if

$$P(skin|rgb) \geq \Theta \quad (4)$$

where $0 \leq \Theta \leq 1$ is a threshold.

In equation 3, $P(skin)$ and $P(\neg skin)$ are the prior probabilities for any color value being skin or non-skin, respectively. Since $P(skin) + P(\neg skin) = 1$, we only need to specify one of these priors. One reasonable choice for the prior probability of skin is the ratio of the total skin pixels in the histogram to the total of all the pixels, i.e.

$$P(skin) = \frac{T_s}{T_s + T_n}.$$

For skin detection, the most important property of equation 4 is the receiver operating characteristic (ROC) curve [21], which shows the relationship between correct detections and false detections as a function of the detection threshold Θ. It turns out that the ROC curve is invariant to the choice of prior $P(skin)$ in the Bayesian model. See the Appendix B for details.

Note that the use of color spaces other than RGB (such as YUV or HSV) will not improve the performance of the skin detector. Detector performance depends entirely on the amount of overlap between the skin and non-skin samples. Colors which occur in both classes with comparable frequencies cannot be classified reliably. No fixed global transformation between color spaces can affect this overlap. On the other hand, color normalization which adjusts the colors in an image based on its global properties could be beneficial in separating skin colors from non-skin colors. We do not employ any form of color normalization because current algorithms do not work well enough over a wide range of images [7].

3.1 Histogram-based Skin Classifier

We conducted a series of experiments with histogram color models using the skin classifier defined by equation 4. For these experiments, we divided our collection of photos into separate training and testing sets. Skin and non-skin color models were constructed from a 6822 photo training set using the procedure described in Section 2.2. In this case there were 4483 training photos which formed the non-skin color model and 2339 training photos which formed the skin color model. From our 6818 photo testing set (4482 non-skin and 2336 skin photos) we obtained two populations of labelled skin and non-skin pixels which were used to test the classifier performance.

Figure 3 shows some examples of skin detection in test images for $\Theta = 0.4$. The classifier does a good job of detecting skin in most of these examples. In particular, the skin labels form dense sets whose shape often resembles that of the true skin pixels. The detector tends to fail on highly saturated or shadowed skin. An example of the former type of failure can be seen on the forehead of the woman in the middle of the top row. An example of the latter failure is visible in the neck of the athlete in the middle of the bottom row.

The example photos also show the performance of the detector on non-skin pixels. In photos such as the house (lower right) or flowers (upper right) the false detections are sparse and scattered. More problematic are images with wood or copper-colored metal such as the kitchen scene (upper left) or railroad tracks (lower left). These photos contain colors which often occur in the skin model and are difficult to discriminate reliably. This results in fairly dense sets of false postives.

Classifier performance can be quantified by computing the ROC curve [21] which measures the threshold-dependent trade-off between misses and false detections. In addition to the threshold setting, classifier performance is also a function of the size of the histogram (number of bins) in the color models. Too few bins results in poor accuracy while too many bins lead to over-fitting.

Figure 4 shows the family of ROC curves produced as the size of the histogram varies from 256 bins/channel to 16. The axis labelled "Probability of correct detection" gives the fraction of pixels labelled as skin that were classified correctly, while "Probability of false detection" gives the fraction of non-skin pixels which are mistakenly classified as skin. These curves were computed from the test data. Histogram size 32 gave the best performance, superior to the size 256 model at the larger false detection rates and slightly better than the size 16 model in two places.

The performance of the skin classifier is surprisingly good considering the unconstrained nature of web images. The best classifier (size 32) can detect roughly 80% of skin pixels with a false positive rate of 8.5%, or 90% correct detections with 14.2% false positives. Its equal error rate is 88%. This corresponds to the point on the ROC curve where the probability of false rejection (which is one minus the probability of correct detection) equals the probability of false detection.

In addition to histogram size, classifier performance is also affected by the amount of training data. This effect is illustrated in Figure 5 (a). We tested the performance of the skin classifier as the amount of training data was increased. We used a 256^3 histogram model for these tests. To do this we took the list of skin and non-skin images in the training set and divided then into chunks containing approximately 2.5 million skin

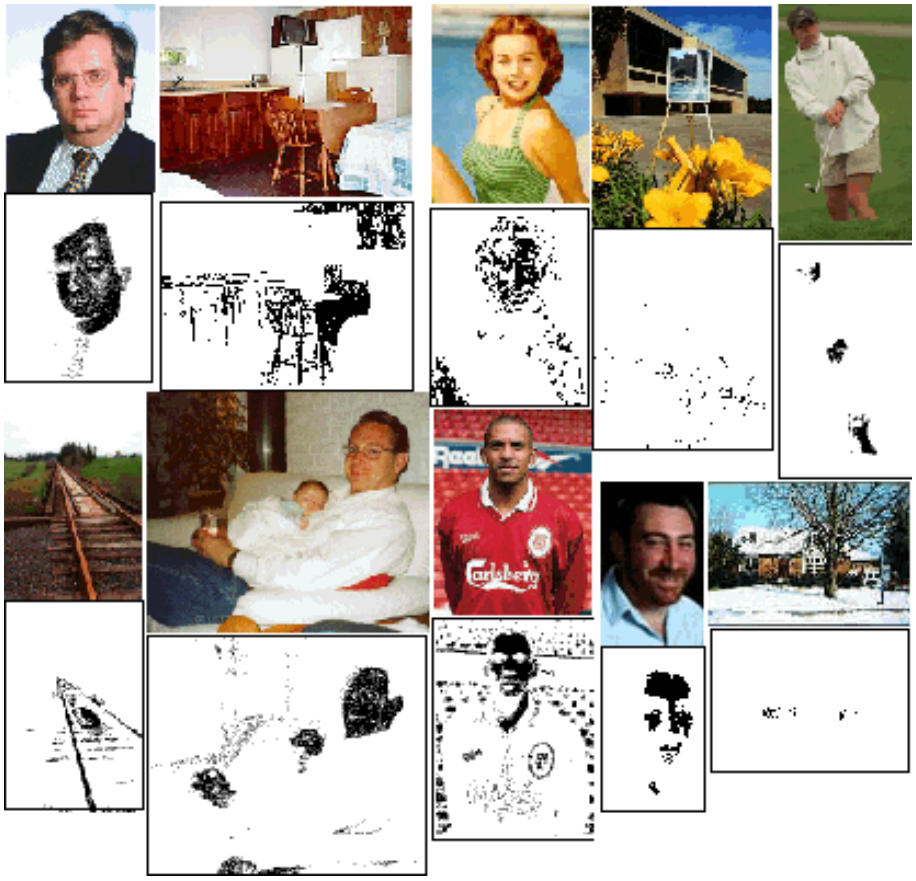


Figure 3: **Examples of skin detections. For each pair, the original image is shown above and the detected skin pixels are shown below.**

pixels and 28 million non-skin pixels. On each iteration we added one such chunk of new skin and non-skin pixels to the evolving training set. A ROC curve was computed at each iteration showing the classifier performance on the partial training set as well as on the full test set.

Figure 5 (a) shows that as more data is added, performance on the training set decreases because the overlap between skin and non-skin data increases. Performance on the test set improves because the test and training distributions become more similar as the amount of training data increases. Performance on both training and test sets converges relatively quickly. There is little change in either after about 8 iterations.

This ROC curve convergence guided our data collection process. During this research, we added photos selected at random from a larger set to our model until we judged that the ROC curves had converged. Our final total of 13,640 photos corresponds to this stopping point. Figure 5 suggests that adding additional photos to our

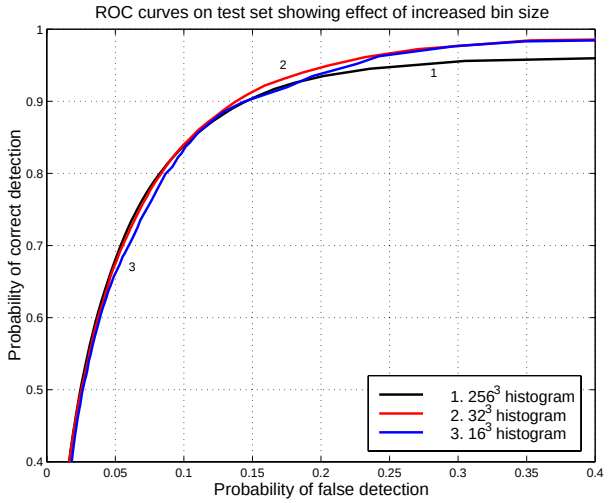


Figure 4: **ROC curves for the skin detector as a function of histogram size.**

model is unlikely to improve the classifier performance.

In our final histogram experiment, we tested the performance of the models trained on a small set of data sampled according to the distribution of skin and non-skin colors in the full training set. We sampled 387,172 skin pixels and 4,261,703 non-skin pixels (1% of the training data) and built histogram models from these samples. In this case we tried histograms with different numbers of bins in order to find the optimal histogram size. The performance is shown in Figure 5 (b). It is almost as good as the histogram model using the full training set. This demonstrates that while a large data set is necessary to capture the underlying distribution of skin and non-skin colors, it is sufficient to train models on a smaller set of samples.

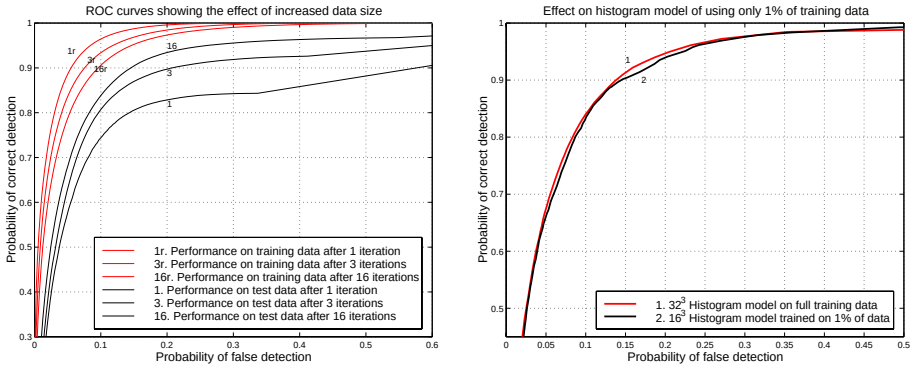
3.2 Comparison to Mixture of Gaussian Classifier

Much of the previous work on skin classification has used a mixture of gaussian model of skin color (some representative examples are [9, 17]). One attraction of mixture models is that they can be made to generalize well on small amounts of training data. We trained mixture models for our dataset and compared their classification performance to the histogram models of Section 3.1.

A mixture density function is expressed as the sum of gaussian kernels as follows:

$$P(\mathbf{x}) = \sum_{i=1}^{N} w_i \frac{1}{(2\pi)^{\frac{3}{2}}|\Sigma_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\mu_i)^T \mathbf{\Sigma}_i^{-1}(\mathbf{x}-\mu_i)}, \tag{5}$$

where $\mathbf{x}$ is a three dimensional RGB color vector and the contribution of the i^{th} gaussian is determined by a scalar weight w_i, mean vector μ_i, and diagonal covariance matrix $\mathbf{\Sigma}_i$.



(a) ROC curves for training and testing as a function of the amount of training data.

(b) ROC curve based on 1 % of the training data shown relative to the full data.

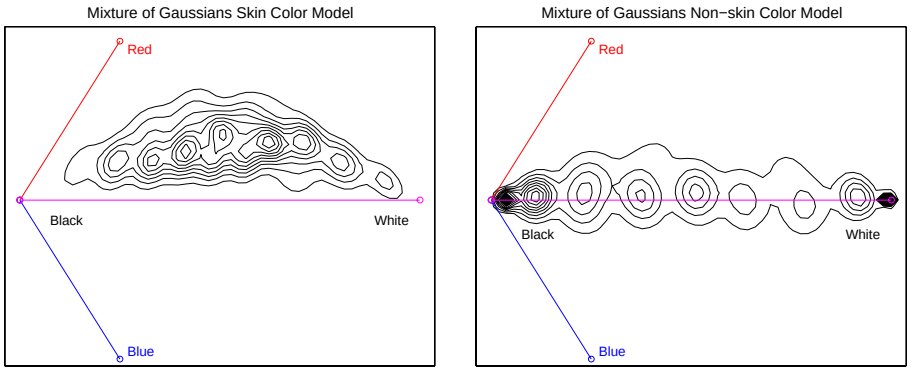
Figure 5: ROC curves for the skin classifier based on histogram color models. The histogram size is 256 in (a) and varies in (b).

We trained two separate mixture models for the skin and non-skin classes. We used 16 gaussians in each model. The models were trained using a parallel implementation of the standard EM algorithm [15]. The non-skin model was trained using the same data as the histogram model in Section 3.1. The skin model was trained using a subset of approximately 74% of the histogram training data. This was simply because that was all the skin training data we had at the time that we performed the mixture experiments.

Contour plots for the mixture of gaussian skin and non-skin models are shown in Figure 6. In both plots the 3-D density is integrated along the green-magenta axis. These plots correspond to the marginalizations of the related histogram models shown in Figures 2(a) and (c). The positions of individual Gaussian kernels can be observed in the level sets.

Figure 7 (a) shows the ROC curve for the skin pixel classifier based on the mixture of gaussian color models. It is shown in comparison to the *best* histogram model ROC curve, which uses a histogram of size 32. We can see that the histogram model gives slightly better performance in this case. This may be somewhat surprising, since skin colors might be expected to form a compact distribution in color space which would presumably be well-suited to mixture density modeling. We can think of two explanations: First, illumination and color quantization effects may perturb the compactness of a "normalized" skin color model. Second, the non-skin model is much less likely to be compact and will have a significant impact on classifier performance.

It is interesting to compare the mixture and histogram models from the standpoint of computational and storage costs. The mixture of gaussian models took significantly longer to train than the histogram models. It took about 24 hours to train both skin and non-skin mixture models using 10 Alpha workstations in parallel. In contrast, the



(a) Contour plot for skin model.

(b) Contour plot for non-skin model.

Figure 6: Contour plots for marginalizations of the mixture of Gaussian skin and non-skin color models.

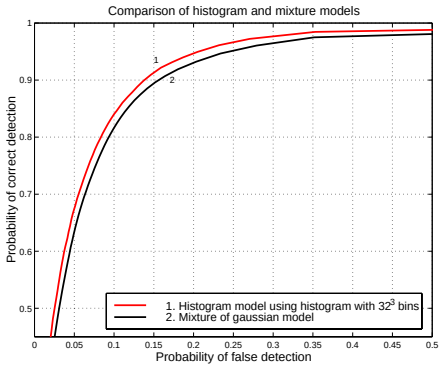
histogram models could be constructed in a matter of minutes. The mixture model is also slower to use during classification since all of the gaussians must be evaluated in computing the probablity of a single color value. In contrast, use of the histogram model results in a fast classifier since only two table lookups are required to compute the probability of skin.

From the standpoint of storage space, however, the mixture model is a much more compact representation of the data. There are a total of 224 floating point parameters (896 bytes assuming 4 byte floats) in the skin and non-skin mixture densities that we used. In contrast, the size 32 histogram model requires 262 Kbytes of storage, assuming one 4 byte integer per bin.

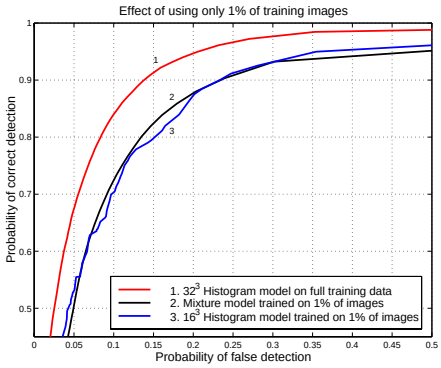
We conducted an additional experiment to verify the importance of having a large data set in obtaining good classifier performance. Since the ROC curves in Figure 5 (a) used a histogram of size 256, there remained the possibility that a model with better generalization, such as a mixture density, might require far less data. To test this hypothesis, we built histogram and mixture models from a much smaller set of images. We picked 30 skin images and 58 non-skin images, which make up approximately 1% of the training set. This sample yielded 406,135 skin pixels and 4,017,896 non-skin pixels for training the models. The ROC curves for the best histogram and mixture models are shown in Figure 7 (b). They both perform much worse than models using the full training set.

3.3 Building Separate Models for Different Image Classes

The skin model described in Section 2.2 learns a single distribution for each of the skin and non-skin color classes. Since the images we are using come from the web, they



(a) ROC curves comparing histogram and mixture models on full training data.



(b) ROC curves for 1 % of images, histogram and mixture models.

Figure 7: ROC curves comparing mixture model and histogram models under varying training data.

were produced under a wide variety of imaging conditions. In light of this, it might be possible to obtain greater accuracy by splitting the images into classes and learning a separate skin and non-skin distribution for each class. The challenge is to decide what the classes are and to determine which class a particular image belongs to.

We have explored an unsupervised learning approach to creating image classes. We computed a number of measures for each image and then created classes through clustering in this feature space. The hope is that if, for example, we could automatically distinguish between bright, high-resolution images and dim, blurry images, then we might get better results using separate color models for these two classes.

The five image measures we used are:

- Average image brightness, where brightness is measured as $(r + g + b)/3$
- Variance of the brightness
- Average distance to the gray axis
- Average gradient energy (average of $\sqrt{dx^2 + dy^2}$)
- Percentage of white and black pixels in the image.

These measures were computed for the same training images used in Section 2.2. Seven clusters were found in the five dimensional space of measures by using the k-means algorithm [5] (we followed the implementation described in [13]). Separate skin and non-skin histograms were built for each cluster. Each image in the training set was assigned to the closest cluster. We used a histogram size of 32 for all of the skin and non-skin models.

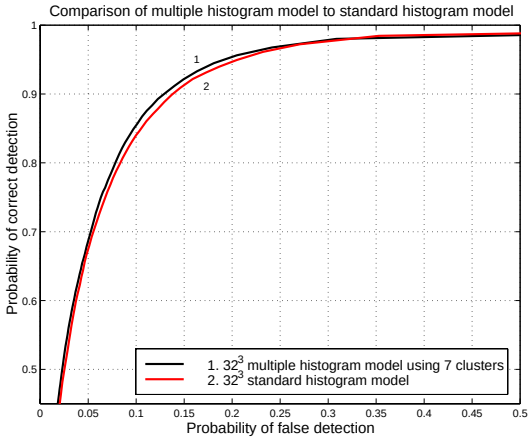


Figure 8: **ROC curves on the test data for the multiple histogram model using 7 clusters and the standard histogram model from Figure 4. All histograms are size 32 (**32^3 **bins).**

Figure 8 shows the ROC curve on the test data for this multiple histogram model using 7 clusters. It is shown in comparison to the ROC curve for the standard (size 32) histogram model using a single distribution for all of the data. This figure shows that the cluster model has slightly better accuracy than the standard model, but the improvement is small. It might be possible to further improve these results by using other feature or by finding better clusters. However, we suspect that we are near the limit of accuracy in predicting skin based solely on a single pixel's color.

3.4 Discussion

We have demonstrated that a surprisingly effective skin detector for web images can be constructed from histogram color models. A surprisingly high equal error rate of 88% was obtained from a histogram of size 32. As Figure 4 demonstrates, this histogram size gave the best generalization. We compared this model to mixture models trained on similar data and found that the histogram model gave superior performance (see Figure 7(a)). This may be somewhat surprising since mixture models are currently a popular choice for color modeling.

We also explored the sensitivity of our detector to the amount of training data. As demonstrated in Figure 5(a), the size of our dataset was determined by monitoring the apparent convergence of the skin detector ROC curve as data was added to the model. This graph suggests that the use of additional training data beyond our current dataset is unlikely to improve the skin detector's performance. We demonstrated in Figure 7(b) that using a smaller amount of photos leads to decreased performance even with color models that peform significant generalization.

Finally, in Section 3.3 we explored the construction of separate color models for

different image classes. The minor improvement that resulted (see Figure 8) suggests that simple global image statistics are not particularly useful indicators of the illumination effects that compromise skin detection.

4 Image Classification by Skin Detection

One interesting application of skin detection is as part of a larger system for detecting people in photos. A person detector that worked reliably on web images could be a valuable tool for image search services such as the AltaVista Photo Finder [5], as well as for image categorization. Our goal in this section is to determine how effective skin color alone can be in this task. We examine the problem of person detection in Section 4.1 and the easier problem of naked person detection in Section 4.2. We find that skin color in the absence of complex shape or texture cues is surprisingly effective for the latter task.

4.1 Person Detection

If the skin detector had no false positives then we could detect people in an image by simply determining if any of the pixels were skin colored. Since the skin detector is not perfect, we have to examine the aggregate properties of the detector output. more clever. We have computed a simple feature vector from the output of the skin detector and then trained a classifier on these features to predict whether a person is present or not. The features we used are:

- Percentage of pixels detected as skin
- Average probability of the skin pixels
- Size in pixels of the largest connected component of skin
- Number of connected components of skin
- Percent of colors with no entries in the skin and non-skin histograms

These features can all be computed in a single pass over the input image. No effort was spent trying to find "optimal" features, so it is quite possible that other features exist that might lead to better performance.

We used 4999 images which were manually classified into person and non-person sets to train a decision tree classifier using C4.5 [14]. The resulting classifier was then tested on a set of 1909 test images. Table 2 summarizes the results.

The results show that simply analyzing color values allows reasonably good classification of images into those containing people and those not, but this cue alone is not sufficient to fully solve the problem of person detection. One obvious problem is that people will expose varying amounts of skin in a given image. Another problem is that many non-skin surfaces exhibit the same color values as skin. Using other cues such as texture and shape would probably lead to greater accuracy, see [12] for a recent example.

[5] `http://image.altavista.com`

	% correctly classified person images	*% correctly classified non-person images*	*Overall % correctly classified images*
Training data	83.0% (2488/2999)	70.6% (1412/2000)	78.0% (3900/4999)
Test data	83.2% (835/1004)	71.3% (645/905)	77.5% (1480/1909)

Table 2: **Performance of person detector on training and test data.**

4.2 Adult Image Detection

By taking advantage of the fact that there is a strong correlation between images with large patches of skin and adult or pornographic images, the skin detector can also be used as the basis for an adult image detector. There is a growing industry aimed at filtering and blocking adult content from web indexes and browsers. Some representative companies are *NetNanny* and *SurfWatch*.[6] All of these services currently operate by maintaining lists of objectionable URL's and newsgroups and require constant manual updating. An image-based scheme has the potential advantage of applying equally to all images without the need for updating (see [6] for additional discussion.)

To detect adult images, we followed the same approach as with person detection. A feature vector based on the output of the skin detector was computed for each training image. The feature vectors included the same 5 features used for person detection plus two new components for the height and width of the image. These two were added based on informal observations that adult images are often sized to frame a standing or reclining figure.

We used 10679 images which were manually classified into adult and non-adult sets to train a neural network classifier. There were 5453 adult images and 5226 non-adult images. The neural network outputs a number between 0 and 1, with 1 indicating an adult image. We can threshold this value to make a binary decision. By varying the threshold, we get the ROC curve shown in Figure 9 for the training data.

To test the adult image detector, we gathered images from two new crawls of the web. Crawl A used adult sites as starting points for the crawl and so gathered many adult images. Crawl B used non-adult sites as starting points and gathered very few adult images. Crawl A consisted of 2365 html pages containing 5241 adult images and 6082 non-adult images (including icons and other graphics). Crawl B consisted of 2692 html pages containing 3 adult images and 13970 non-adult images. We used the adult images from Crawl A and the non-adult images from Crawl B to test the classifier.

The ROC curve for the adult image detector on the test set is shown in Figure 9. The detector achieved, for example, 85.8% correct detections with 7.5% false positives. This performance is suprisingly good considering the simple color-based features that were used. In most previous systems for person or face detection, skin color is used as kind of prefilter to guide a detection process which is based wholely on shape, texture, etc. These results suggest that skin color may actually deserve a more prominent role in deciding whether or not a person is present in an image. A direct comparison between our adult detector and others can be found in Section 5.

[6] *http://www.netnanny.com* and *http://www.surfwatch.com*.

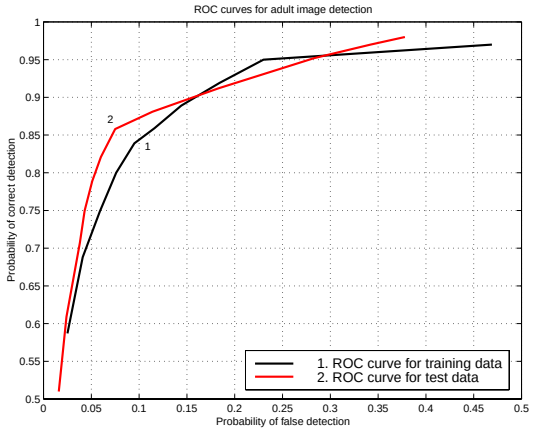


Figure 9: **ROC curves for the adult image detector on both training and testing images**

	% correct detections	*% false alarms*
Color-based Detector	85.8%	7.5%
Text-based Detector	84.9%	1.1%
Combined Detector	93.9%	8.0%

Table 3: **Comparison of adult image detector using color-based, text-based and combined classifiers on the test data**

4.3 Incorporating Text Features in Adult Image Detection

We have also explored combining the adult image detector just described with a text-based classifier which uses the text occurring on a web page with an image to determine if an image is pornographic. The text-based detector classifies whole web pages as adult or not. To apply the text classfier to each individual image occurring on a page, we simply use the global label for the page with each image it contains. The text-based classifier on its own achieves 84.9% correct detections with 1.1% false positives. There is no threshold associated with the text-based classifier we tested, so only one point on the ROC curve is realized.

We combined the color-based detector (using a threshold that yielded 85.8% correct detections and 7.5% false positives) with the text-based detector by using an "OR" of the two classifiers, i.e. an image is labelled adult if either classifier labels it adult. The combined detector correctly labels 93.9% of the adult images from crawl A and obtains 8% false positives on the non-adult images from crawl B. Table 3 summarizes these results.

The results show that simply analyzing color values allows very good detection of adult images. Not surprisingly, adding information from the surrounding text can boost performance significantly.

5 Previous work

While there has been much previous work on skin color modeling, we know of no previous effort based on such a large corpus of training and testing data and no comparably detailed study of skin classification in web images.

Many systems for tracking or detecting people in user-interface or video-conferencing applications have employed skin color models. Histogram models are employed by Schiele and Waibel [17] and Kjeldsen and Kender [10]. Yang *et al.* [25] model skin color as a single Gaussian, while Jebara *et al.* [9] employ a mixture density. In all of these systems, the color model is trained on a small number of example images taken under a representative set of illumination conditions. Most, with the exception of [10, 9], do not use non-skin models. These color models are effective in the context of a larger system, but they do not address the question of building a global skin model which can be applied to a large set of images.

The closest works to ours are two systems for detecting images containing naked people developed by Forsyth *et al.* [6] and Wang *et al.* [24]. Both of these systems use a skin color model as a preprocessing step and have been tested on a corpus of web images. The skin color model used by Forsyth *et al.* consists of a manually specified region in a log-opponent color space. Detected regions of skin pixels form the input to a geometric filter based on body plans. The WIPE system developed by Wang *et al.* uses a manually-specified color histogram model as a prefilter in an analysis pipeline. Input images whose average probability of skin is low are rejected as non-offensive. Images that contain skin pass on to a final stage of analysis where they are classified using wavelet features. Since neither of these works report the performance of their skin detector in isolation, a direct comparison with Figure 4 is not possible.

We can compare the overall effectiveness of the Forsyth and WIPE systems in detecting naked people to our detector, which is described in Section 4.2. In contrast to this previous work, our detector uses very weak global attributes of the detected skin pixels to classify the image. Both body plans and wavelet coefficients have more descriptive power than our seven element feature vector. Perhaps surprisingly, we find that our detection performance is comparable to theirs.

Forsyth reports two sets of experimental results: the skin filter alone, and used in conjunction with the geometric filter. Their skin filter is not directly comparable to ours, as it uses texture analysis and groups pixels into skin regions. However, they also report surprisingly strong performance when images that contain one or more detected skin regions are labelled as containing naked people. The detection rate is 79.3 % with a false alarm rate of 11.3 %. When combined with the geometry filter the false positives fall to 4.2 % while the detection rate falls to 42.7 % for the "primary" configuration of the system.

Wang *et al.* report the overall the results of the WIPE system on objectionable images: 96% detection rate with 9% false positives. Table 4 gives a summary of the performance of the two systems in comparison to ours. The Forsyth test set contained 4854 images, the WIPE test set contained 11,885 images, and our test contained 19,211 images.

It is hard to draw strong conclusions from the comparison in Table 4, since the testing sets for all three systems are completely different and they each exploit different

System	*Detection Rate*	*False Alarm Rate*
Forsyth (Skin Only)	79.3	11.3
Jones-Rehg	88.0	11.3
Forsyth (Skin+Geom)	42.7	4.2
Jones-Rehg	75.0	4.2
WIPE	96	9.0
Jones-Rehg	86.7	9.0

Table 4: Performance comparison for three adult image detection systems.

image cues. Since our system is fairly weak in exploiting shape or geometry cues, we feel it is a fairly representative test of the value of color information alone. These results suggest that adult detection systems can get more mileage out of skin color than might have been expected.

6 Conclusions

The existence of large image datasets such as the set of photos on the world wide web make it possible to build powerful generic models for low-level image attributes like color using simple histogram learning techniques. We have demonstrated this point empirically by constructing color models for skin and non-skin classes from a dataset of nearly 1 billion labelled pixels. Using visualization techniques we demonstrate a surprising degree of separability between these two classes. A skin detector constructed from these models achieved a detection rate of 80% with 8.5% false positives. We compared histogram models to mixture of Gaussian models on the skin detection task. We found that histogram models perform slightly better and are computationally much faster. We have also constructed a surprisingly effective detector for naked people from the output of our skin classifier. This suggests that skin color can be a more powerful cue for detecting people in unconstrained imagery than was previously suspected.

Acknowledgments

The authors would like to thank Michael Swain for some valuable discussions and Pedro Moreno for his help in fitting the mixture models using a parallel implementation of the EM algorithm. We would also like to thank Nick Whyte of AltaVista for providing the image dataset.

Appendix

A Dataset of Web Images

In this section we provide more details about our image dataset, the segmentation process we used to obtain labeled training pixels, and instructions on how to obtain our data for academic research use. The starting point for our dataset was a large collection of image files we obtained from a parallel crawl of the world wide web from multiple starting points. This image set was pruned by manually removing all files that were not photos.[7] We obtained our dataset of 13,640 photos by sampling randomly from this larger set.

Each photo in our dataset was processed in the following manner: The photo was examined to determine if it contained skin. If no skin was present, it was placed in the non-skin group. If it contained skin, regions of skin pixels were manually labeled using the software tool shown in Figure 10. This tool allows a user to interactively segment regions of skin by controlling a connected-components algorithm. Clicking on a pixel establishs it as a seed for region growing. The threshold slider controls the Euclidean distance in RGB space around the seed that defines the skin region. By clicking on different points in the photo and adjusting the slider, regions of skin with fairly complex shapes can be segmented quickly. In labelling skin we were careful to exclude the eyes, hair, and mouth opening. The result is a binary mask which is stored along with each photo that identifies its skin pixels.

Non-skin pixels that appeared within a photo containing skin were not included in either color model. This was necessary because of the difficulty in getting a perfect segmentation of the skin in any given image. Some photos contained skin patches of such a small size (e.g. crowd scenes) that segmentation was problematic. Even in photos with large regions of skin it was often hard to precisely define their boundaries, (e.g. on the forehead where skin is obscured by hair.) We chose the conservative strategy of segmenting the easily identifiable skin pixels and discarding the remainder to avoid contaminating the non-skin model.

One of the issues that arises in a dataset taken from the web is the question of color quantization. Digital images obtained from different sources such as scanners, capture cards, and digital cameras will have different color resolutions. Unfortunately, most of the information about color resolution is lost once an image has been stored in one of the file formats that are in wide-spread use on the web.

The two most common image encoding schemes used on the web are GIF and JPEG. GIF images are represented by a palette of discrete colors (colormap) which can vary in size. The JPEG File Image Format (JFIF) in which JPEG-encoded images are stored specifies a 24 bit RGB color resolution (see [11] for more details.) In all except the highest quality color images, however, the effective color resolution of the photo is far less than 24 bits. Since it is generally impossible to deduce the true color resolution, we have no choice but to work with the original 24-bit RGB color values.

[7] An automatic approach to distinguishing photos from graphics is described in [1].

Figure 10: **Snap shot of the tool for segmenting the skin region of an image. The left image shows the completed manual segmentation with the skin pixels highlighted in red. The right image shows the original image.**

A.1 Obtaining Our Dataset

We are making our dataset of labelled skin and non-skin photos available to university researchers. Contact Mike Jones at `mjones@crl.dec.com` for instructions.

B Invariance of ROC Curve to Choice of Prior

Equation 3 which gives the formula for $P(skin|rgb)$ includes two prior probabilities: $P(skin)$ and $P(\neg skin)$. The following proof shows that the choice of these priors does not affect the ROC curves for skin detection.

First, we must be precise about how a ROC curve is computed. We need a set of test images which have been segmented into skin and non-skin regions. We can store this test data in skin and non-skin histograms as before. Then the percentage of correct detections and false positives for a given threshold Θ is calculated by summing up separately the number of skin counts and non-skin counts in the test images for each RGB value where $P(skin|rgb) \geq \Theta$ and then dividing by the total skin count and total non-skin count, respectively. To be precise, this is expressed in pseudo-code below which plots the ROC curve given test data stored in skin ($s'[rgb]$) and non-skin ($n'[rgb]$) histograms. These histograms are distinct from the histograms built from training data and used to define the skin model.

$TP = 0$ *(This stores the true positives)*

$FP = 0$ *(This stores the false positives)*
for $0 \leq \Theta \leq 1$ {
  for all RGB values,
    if $P(skin|rgb) \geq \Theta$ {
      $TP = TP + s'[rgb]$
      $FP = FP + n'[rgb]$
    }
  $TP = TP/T_s$ *(T_s is the total skin pixels in the test images)*
  $FP = FP/T_n$ *(T_n is the total non-skin pixels in the test images)*
  plot point FP versus TP
}

Now we can state and prove the following theorem.

Theorem 1 *The ROC curve is the same for any choice of priors (greater than 0 and less than 1) in the Bayesian skin model.*

Proof

We will use the notation $P_{p1}(x)$ to mean the probability of x assuming prior p1.

The following three statements imply the theorem:

- if $P_{p1}(skin|rgb_1) = P_{p1}(skin|rgb_2)$ then $P_{p2}(skin|rgb_1) = P_{p2}(skin|rgb_2)$
- if $P_{p1}(skin|rgb_1) > P_{p1}(skin|rgb_2)$ then $P_{p2}(skin|rgb_1) > P_{p2}(skin|rgb_2)$
- if $P_{p1}(skin|rgb_1) < P_{p1}(skin|rgb_2)$ then $P_{p2}(skin|rgb_1) < P_{p2}(skin|rgb_2)$

where rgb_1 and rgb_2 are different rgb color values.

If these three statements are true then the ROC curve is the same for any choice of priors. One can see this by looking at figure 11 which shows a hypothetical histogram of probabilities for a few RGB values. Every point on the ROC curve is computed by drawing a line horizontally across the histogram which corresponds to a threshold. The ROC curve only depends on which set of RGB values are labelled skin and which are not. It does not matter if the heights of the lines in figure 11 change as long as the relative heights are preserved. In other words, if the histogram were to change due to a new prior being chosen, then if the relative probabilities do not change with respect to each other the ROC curve will be the same. The three conditions above state that if two RGB values have equal probability assuming prior $p1$ then they will also have equal probability assuming prior $p2$. If one RGB value is greater than another assuming prior $p1$ then that RGB value will also be greater than the other assuming prior $p2$. The "less than" case is equivalent to the greater than case.

So, if we can prove these three statements then we have proven the theorem. First, assume

$$P_{p1}(skin|rgb_1) = P_{p1}(skin|rgb_2)$$

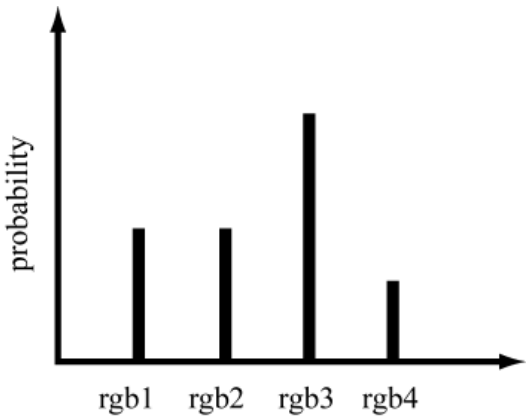


Figure 11: **Graph of the probabilities of four hypothetical RGB values used to illustrate the idea of the proof. The first two RGB values have the same probability, the third has greater probability and the fourth has smaller probability.**

$$\Rightarrow \frac{P(rgb_1|skin)p_1}{P(rgb_1|skin)p_1 + P(rgb_1|\neg skin)(1-p_1)} = \frac{P(rgb_2|skin)p_1}{P(rgb_2|skin)p_1 + P(rgb_2|\neg skin)(1-p_1)}$$

$$\Rightarrow \frac{1}{1 + \frac{P(rgb_1|\neg skin)(1-p_1)}{P(rgb_1|skin)p_1}} = \frac{1}{1 + \frac{P(rgb_2|\neg skin)(1-p_1)}{P(rgb_2|skin)p_1}}$$

$$\Rightarrow \frac{P(rgb_1|\neg skin)(1-p_1)}{P(rgb_1|skin)p_1} = \frac{P(rgb_2|\neg skin)(1-p_1)}{P(rgb_2|skin)p_1}$$

$$\Rightarrow \frac{P(rgb_1|\neg skin)}{P(rgb_1|skin)} = \frac{P(rgb_2|\neg skin)}{P(rgb_2|skin)}$$

Therefore, since $\frac{1-p_1}{p_1}$ factors out of both sides of the equation, we could replace it with $\frac{1-p_2}{p_2}$ and still have a true statement. Thus, the prior can be changed without changing the veracity of the assumption.

This implies

$$P_{p2}(skin|rgb_1) = P_{p2}(skin|rgb_2)$$

assuming $P_{p1}(skin|rgb_1) = P_{p1}(skin|rgb_2)$. This proves the first statement.

The second and third statements can be proved in the same manner. Simply replace $=$ with $>$ or $<$ in the assumption and carry through the same analysis.

Since all three statements are true, we have proven that the ROC curve is the same for any choice of priors in the Bayesian skin model.

C Facts About the Bayesian Model

If we choose the prior

$$P(skin) = \frac{T_s}{T_s + T_n}$$

and substitute this into Bayes rule (along with $P(rgb|skin) = \frac{s[rgb]}{T_s}$ then after some algebraic simplifications, we end up with

$$P(skin|rgb) = \frac{s[rgb]}{s[rgb] + n[rgb]}.$$

This says that the probability of skin at a particular RGB value is the ratio of the skin count at that RGB value over the total skin and non-skin counts at that value.

Another interesting point is that using the criterion

$$P(rgb|skin) \geq P(rgb|non\_skin)$$

corresponds to using the Bayesian model with a particular threshold.

To see this we use the fact we just showed which is that

$$\frac{s[rgb]}{s[rgb] + n[rgb]} \geq \Theta$$

is a Bayes decision criteria. Consider the choice of threshold $\Theta = \frac{T_s}{T_s + T_n}$. This gives us

$$\begin{aligned}
\frac{s[rgb]}{s[rgb] + n[rgb]} &\geq \frac{T_s}{T_s + T_n} \\
\Rightarrow s[rgb] &\geq \frac{T_s s[rgb] + T_s n[rgb]}{T_s + T_n} \\
\Rightarrow (1 - \frac{T_s}{T_s + T_n}) s[rgb] &\geq \frac{T_s}{T_s + T_n} n[rgb] \\
\Rightarrow \frac{T_n}{T_s + T_n} s[rgb] &\geq \frac{T_s}{T_s + T_n} n[rgb] \\
\Rightarrow \frac{s[rgb]}{T_s} &\geq \frac{n[rgb]}{T_n} \\
\Rightarrow P(rgb|skin) &\geq P(rgb|non_skin)
\end{aligned}$$

Thus, using the criteria $P(rgb|skin) \geq P(rgb|non\_skin)$ to classify pixels as skin is equivalent to the Bayesian model with a particular threshold.

References

[1] Vassilis Athitsos, Michael J. Swain, and Charles Frankel. Distinguishing photographs and graphics on the world wide web. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 10–17, San Juan, Puerto Rico, June 20 1997.

[2] Corbis. Internet photo store. URL: *http://www.corbis.com*, 1998.

[3] Jeremy S. De Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images. In *Proceedings of SIGGRAPH 97*, pages 361–368, Los Angeles, CA, August 3-8 1997.

[4] Jeremy S. De Bonet and Paul Viola. Texture recognition using a non-parametric multi-scale statistical model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 641–647, Santa Barbara, CA, June 23-25 1998.

[5] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[6] David A. Forsyth, Margaret Fleck, and Chris Bregler. Finding naked people. *International Journal of Computer Vision*, 1996. Under review. Available at `http://www.cs.berkeley.edu/ daf/newo2.ps.Z`.

[7] Brian Funt, Kobus Barnard, and Lindsay Martin. Is machine color constancy good enough? In Hans Burkhardt and Bernd Neumann, editors, *Proceedings of 5th European Conference on Computer Vision*, volume I, pages 445–459, Freiburg, Germany, June 1998.

[8] David J. Heeger and James R. Bergen. Pyramid-based texture analysis/synthesis. In *Proceedings of SIGGRAPH 95*, pages 229–238, Los Angeles, CA, August 6-11 1995.

[9] Tony S. Jebara and Alex Pentland. Parameterized structure from motion for 3d adaptive feedback tracking of faces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 144–150, San Juan, Puerto Rico, June 17-19 1997.

[10] Rick Kjeldsen and John Kender. Finding skin in color images. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, pages 312–317, Killington, VT, October 14-16 1996.

[11] James D. Murray, Deborah Russell, and William Vanryper. *Encyclopedia of Graphics File Formats*. O'Reilly and Associates, 2nd edition, 1996.

[12] Michael Oren, Constantine Papageorgiou, Pawan Sinha, Edgar Osuna, and Tomaso Poggio. Pedestrian detection using wavelet templates. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 193–199, San Juan, Puerto Rico, June 17-19 1997.

[13] Kris Popat and Rosalind W. Picard. Cluster-based probability model and its application to image and texture processing. *IEEE Transactions on Image Processing*, 6(2):268–284, 1997.

[14] J. Ross Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kauffman Publishers, Inc., 1993.

[15] R. Redner and H. Walker. Mixture densities, maximum likelihood, and the EM algorithm. *SIAM Revew*, 26:195–239, 1994.

[16] Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, January 1998.

[17] Bernt Schiele and Alex Waibel. Gaze tracking based on face-color. In *Proceedings of the International Workshop on Automatic Face- and Gesture-Recognition*, pages 344–349, Zurich, Switzerland, June 26-28 1995.

[18] Eero P. Simoncelli. Statistical models for images: Compression, restoration and synthesis. In *Proceedings of the 31st Asilomar Conference on Signals, Systems and Computers*, 1997.

[19] Eero P. Simoncelli and Javier Portilla. Texture characterization via joint statistics of wavelet coefficient magnitudes. In *Proceedings of the Fifth International Conference on Image Processing*, 1998.

[20] Michael Swain, 1998. Personal communication.

[21] Harry L. Van Trees. *Detection, Estimation, and Modulation Theory*, volume I. Wiley, 1968.

[22] Gregory K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1), Feb 1992.

[23] Brian A. Wandell. *Foundations of Vision*. Sinauer Associates, Sunderland, MA, 1995.

[24] James Ze Wang, Jia Li, Gio Wiederhold, and Oscar Firschein. System for screening objectionable images using daubechies' wavelets and color histograms. In *Proc. of the International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services*, pages 20–30, 1997.

[25] Jie Yang, Weier Lu, and Alex Waibel. Skin-color modeling and adaptation. In *Proceedings of the 3rd Asian Conference on Computer Vision*, pages 687–694, Hong Kong, China, January 8-10 1998.

[26] Song Chun Zhu and David Mumford. Prior learning and gibbs reaction-diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(11):1236–1250, Nov 1997.

[27] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy. *International Journal of Computer Vision*, 27(2):107–126, March 1998.

COMPAQ

Statistical Color Models with Application to Skin Detection

Michael J. Jones James M. Rehg

CRL 98/11
December 1998