BU CS TR99-015.v2, Dec. 1999 (revised in March 2000).
To appear in **Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2000)**.

Estimation and Prediction of Evolving Color Distributions for Skin Segmentation Under Varying Illumination

Leonid Sigal, Stan Sclaroff, and Vassilis Athitsos
Image and Video Computing Group - Computer Science Dept.
Boston University - Boston, MA 02215

Abstract

A novel approach for real-time skin segmentation in video sequences is described. The approach enables reliable skin segmentation despite wide variation in illumination during tracking. An explicit second order Markov model is used to predict evolution of the skin color (HSV) histogram over time. Histograms are dynamically updated based on feedback from the current segmentation and based on predictions of the Markov model. The evolution of the skin color distribution at each frame is parameterized by translation, scaling and rotation in color space. Consequent changes in geometric parameterization of the distribution are propagated by warping and re-sampling the histogram. The parameters of the discrete-time dynamic Markov model are estimated using Maximum Likelihood Estimation, and also evolve over time. Quantitative evaluation of the method was conducted on labeled ground-truth video sequences taken from popular movies.

1 Introduction

Locating and tracking patches of skin-colored pixels through an image sequence is a tool used in many face recognition and gesture tracking systems [1, 4, 5, 7, 8, 9, 12, 13, 14]. An important challenge of any skin-color tracking system is to accommodate varying illumination conditions that may occur within an image sequence. Some robustness may be achieved via the use of luminance invariant color-spaces [13, 7]; however, this method can withstand only changes that skin-color distributions undergo within a narrow set of conditions.

The conditions that we are concerned with in this paper are broader than those assumed in many previous systems. In particular, we are concerned with three conditions: 1.) time-varying illumination, 2.) multiple sources, with time-varying illumination, and 3.) single or multiple colored sources. Most previous skin segmentation and tracking systems address only condition 1, defined over a narrow range (white light). Nevertheless, conditions 2 and 3 are also important, and have to be addressed in order to build a general purpose skin-color tracker. We will now list a few common scenarios that may lead to consideration of some, all, or a combination of the conditions cited above.

Consider a person driving a car at night. Illumination from street lights and traffic lights will be at least in part responsible for the color appearance of his/her skin. Hence if we want to build a skin color tracking system that would be used in surveiling the driver [10], we need to account for varying illuminant intensity and color.

Skin-color person tracking is also useful in indexing multimedia content such as movies. In this case, multiple colored lights with varying intensity play a direct role, since many movies are filmed with theatrical lighting to dramatize the effects of the screenplay.

Still another example of time-varying color illuminant is apparent in observing a person walking down a corridor with windows or lights that are significantly spaced apart. The color appearance of the person's skin will smoothly change as they move towards and then away from various light sources along the corridor.

Finally, it should be noted that it is not necessary to have colored lights to achieve effects equivalent to those that occur with colored lighting. Equivalent effects commonly arise due to surface inter-reflectance. For instance, consider a person walking down a corridor that has colored walls and/or carpet, or a person wearing colorful clothing. These surfaces reflect a color tinge onto the person's skin.

These are a few examples of applications that motivate our approach. Even though we agree that the majority of everyday lighting effects are due to white light attenuation, we hold that it is important to consider alternatives as well, in order to have a robust skin-color tracker that can handle a wider variety of environmental conditions.

In this paper we propose a new technique that allows for a more general representation of skin-color. An explicit second order Markov model is used to predict evolution of the skin color distribution over time. Histograms are dynamically updated based on feedback from the current segmentation and based on predictions of the Markov model. The parameters of the discrete-time dynamic Markov model are estimated using Maximum Likelihood Estimation, and also evolve over time. Quantitative evaluation of the method was conducted on labeled ground-truth video sequences taken from popular movies, and the results are encouraging.

2 Related Work

In a study of skin-color distributions conducted by Yang and Waibel [13], three major conclusions were found. First, human skin-color distributions are clustered in the chromatic color space; the skin color distribution for a per-

son, regardless of identity or ethnicity, occupies a relatively small area within the color space. Second, skin-color differences among people can be reduced by intensity normalization. Third, under certain lighting conditions, a skin-color distribution can be characterized by a multivariate normal distribution in the normalized color space.

Therefore, under certain conditions the skin-color distribution of each individual can be expressed as a multivariate normal distribution; however, parameters of the distribution can vary significantly with people and lighting conditions. This means that in order to build a system that is general enough to model and track different people, or robust enough to handle even modest variations in illumination conditions, we have to employ an algorithm that would adjust the parameters of the distribution accordingly.

This insight aided Yang and Waibel [13] in the design of their system. Their adaptation technique used a linear combination of previous parameters to estimate the new parameters for the mean and covariance of the multivariate Gaussian distribution. The algorithm was implemented in normalized (r, g) color space, using the Expectation Maximization algorithm (EM).

In a similar real-time human tracking system proposed by Hafner and Munkelt [5], skin color was modeled by a 2D normal distribution in u and v components of Huv color space. Unlike Yang and Waibel, exponential (instead of linear) functions were used in the weighted estimation of the evolving distribution's parameters.

Oliver and Pentland [7] built a real-time system for tracking and classification of human face and lip motion. Spatial coordinates (x, y) were combined with (r, g) normalized color components in a 4D feature vector for each pixel. These features were used as input to an incremental EM algorithm that dynamically estimated the Gaussian mixture models for background and foreground. Pixels were grouped into skin-color blobs. Kalman filters were used to filter spatial parameters for each blob.

Raja, et al., [8] developed a tracking system that employed Gaussian mixtures to model skin, clothes and background. It was assumed that a skin-color distribution can be modeled by a low order Gaussian mixture, where the number of components does not change over time or over a range of conditions. The system's use of the hue-saturation color space made it robust to minor illuminant changes.

In Birchfield's real-time head tracking system [1], the projection of a head in the image plane was modeled by an ellipse. The intensity gradient near the edge of the ellipse and a color histogram representing the interior were used to update the ellipse parameters over time. Use of (B-G, G-R, B+G+R) color space provided robustness to specular highlights, and uniform shifts in white illumination. Use of histograms made it more versatile; however, these histograms were static and did not model varying illumination.

Darrell, *et al.* proposed an integrated approach to real-time person tracking that combined a number of stereo, color, and neural net based approaches for face detection [4]. Unlike other systems described thus far, this system employed stereo views, and hence tended to be more stable and robust to occlusions. The authors empirically found that skin-color can be modeled as a single Gaussian in RGB "log color opponent" space, and employed this representation in their approach.

Name-It, a system for finding faces in newscast video, employed a normalized (r, g) space for skin color tracking [9]. A single Gaussian model with standard Bayesian classifier was used for skin/non-skin pixel classification. An eigenvector based technique was used to detect faces.

In summary, techniques that adapt the color distribution over time perform much better. All systems employ a color space representation that provides robustness to illumination variation. Some systems add shape or blob constraints to further improve tracking. We propose a system that goes even further, by employing *predictive adaptation* in modeling the color distribution over time. As will be seen, accurate predictions can lead to a better segmentation under varying illumination conditions.

3 Overview of Approach

The goal is to track a moving skin-color distribution as defined by an adaptive color histogram in color space. Tracking is done by predicting the future parameters of the distribution and applying a warping on the distribution based on those predictions. The algorithm has three stages: initialization, learning, and then steady-state prediction/tracking.

The initialization stage segments the first frame of the image sequence to give an initial estimate for the skin-color distribution to be tracked. This is done by using a two-class Bayes' classifier. The prior histograms used for classification are precomputed off-line using the database provided by Jones and Rehg [6]. The resulting crude estimate is then refined with binary image processing. The final result of the initialization phase is the binary mask for the skin color regions to be tracked.

The learning stage uses an EM process over the first few frames in the video sequence. At each frame, the estimation step is histogram-based segmentation and the maximization step is histogram adaptation. This process defines the evolution of the distribution in discrete time. The evolution of the distribution is implicitly defined in terms of translation, rotation, and scaling of the samples in color space. The transformation parameters are easily estimated via standard statistical methods. Given the evolution of parameters, we can estimate the motion model for the distribution, and hence predict further deformations. The motion model that we use for the predictions is a second order discrete-time Markov model. The Markov model parameters are estimated by maximum likelihood estimation.

Once a motion model is learned we proceed to the prediction/tracking stage. At this stage, in addition to segmentation and distribution estimation, changes in translation,

scaling and rotation of the distribution are *predicted* given the Markov model estimated in the learning stage. The parameters of Markov model are re-estimated over time as well. By predicting parametric changes, we can get a better estimate of the true distribution at the next time step. Even though adaptive histograms are used for segmentation, we cannot apply the predictions to the histograms directly due to the problems with resolution and sampling. Instead the predictions are propagated via a transformation applied on the samples directly. The newly transformed samples are used to estimate the histogram at the next frame.

Each of the three basic stages of the algorithm will now be described in greater detail.

4 Initialization

The first stage of the system is designed to give an initial estimate for the location of the foreground (skin) and background (non-skin) regions in the first frame of the image sequence. This is achieved by segmenting the first frame, with histogram-based conditional probability distributions for the two classes that have been obtained off-line.

4.1 Prior Histogram Learning

Histograms for the skin and background distributions are learned off-line from a database provided by Jones and Rehg [6]. The database contains 4675 skin images with corresponding masks and 8965 non-skin images. All images were collected from the world wide web and skin regions were labeled by hand.

Following [6], histogram-based distributions were computed at a $32 \times 32 \times 32$ bin resolution in RGB color space. Results obtained in [6] showed that $32 \times 32 \times 32$ bin histograms are not only sufficient but are superior in the segmentation to the fully-ranked $256 \times 256 \times 256$ histograms. Conditional probability densities were obtained by dividing the count of pixels in each histogram bin by the total number of pixels in the histogram. The conditional densities will be denoted $P(rgb|fg)$, and $P(rgb|bg)$,where fg denotes foreground, bg background, and $rgb \in \Re^3$.

4.2 Skin Segmentation Using Prior Histograms

Using Bayes' formula, we can compute $P(fg|rgb)$ and $P(bg|rgb)$. The classification boundary can be drawn where the the ratio of $P(fg|rgb)$ and $P(bg|rgb)$ exceeds some threshold K that is based on a relative risk factor associated with misclassification. For example

$$K < \frac{P(fg|rgb)}{P(bg|rgb)} = \frac{P(rgb|fg)P(fg)}{P(rgb|bg)P(bg)} \qquad (1)$$

corresponds to pixel value rgb being labeled as foreground. Rearranging terms

$$K \times \frac{1 - P(fg)}{P(fg)} < \frac{P(rgb|fg)}{P(rgb|bg)}, \qquad (2)$$

where $P(fg)$ is the probability of an arbitrary pixel in an image being skin. Clearly this probability will vary from image to image, but given a large enough data set we can come up with the aggregate probability that can serve as our best estimate. In our training database, $P(fg) = 0.09$.

Given $P(fg)$, we can now empirically establish the threshold K. One of the standard ways of determining the threshold is by computing a Receiver Operating Characteristic (ROC) curve. The ROC curves in Fig. 1 show the trade off between the true positives and false positives for various possible settings of the decision criterion K.

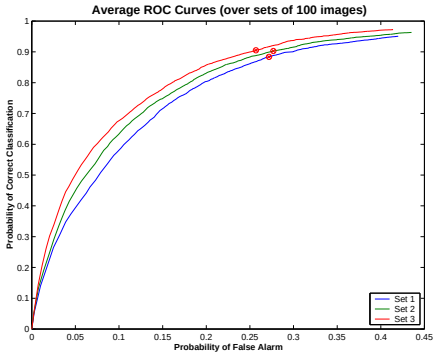

Figure 1: Average ROC curves (computed over three random sets of 100 images excluded from training data) for skin segmentation as a function of threshold K. The x-axis corresponds to the probability of false detection, and the y-axis to the probability of correct classification.

A threshold was chosen such that at least 85% correct classification is achieved while having under 25% chance of false alarm. This choice was made in light of [6] and the fact that the optimal value for the threshold should lay near the bend of the ROC curve. The selected threshold was $K = 0.06$. This was consistent across a number of trials.

The result of the pixel classification scheme above is a binary image mask in which 0's correspond to background pixels, and 1's to foreground pixels. In order to minimize noise effects, we employ size and hole filtering before the binary mask is passed to the learning stage of the system.

5 Learning

Thus far, only aggregate statistics have been employed in segmentation. However, our ultimate goal is to learn the statistics that are specific to the image sequence at hand. The mask for the first frame of the sequence (provided by the initialization) is a good initial estimate of skin and non-skin regions. The pixels from those regions can be used to re-estimate histograms for foreground and background. The new histograms are sequence-specific, and hence are better estimates. The new sequence-specific value for the $P(fg)$ is also re-estimated based on the image mask.

However, using static histograms for the image sequence in which the distribution constantly changes is inappropriate; hence, we employ an adaptive histogram scheme to

facilitate the tracking of the distributions. From the foreground and background distributions observed over an initial sequence of frames, sequence-specific motion patterns are learned. A second-order Markov process is used to model evolution of the color distributions over time. The formulation will now be described in greater detail.

5.1 Color Space for Skin-Color Tracking

An important aspect of any skin-color tracking system is choosing a color space that is relatively invariant to minor illuminant changes. The two most popular color spaces that have proved to be robust to minor illuminant changes are HSV and normalized RGB. In preliminary experiments we found that HSV color space is much better suited than normalized (r, g) for estimation and prediction of skin-color distribution evolution in image sequences taken from entertainment videos and movies.

The only disadvantage of the HSV color space is the costly conversion from standard RGB source. We handled this problem by quantizing the HSV space into (64x64x64) RGB to HSV lookup table. To gain a uniform sampling of the color space, each of the HSV color channels is normalized to floating point values between 0 and 1, given the expected range of HSV values.

5.2 Histogram Representation of Distributions

Skin color, even though clustered in space, cannot be adequately represented as a single Gaussian in general. The mixture of Gaussians representation is much more powerful. With a small number of mixtures, evaluation and updates of the probability density function can be done in real time. However, as soon as more mixtures are needed for representation, this approach becomes infeasible.

One major advantage of the histogram representation is that the probability density function can be evaluated trivially regardless of the complexity of the underlying distribution. Another advantage is histogram generality. The main disadvantage is that histograms in general are bad for representing sparse data, where only a fraction of the necessary samples is available. This can be dealt with via interpolation or Gaussian filtering of the histogram.

We will assume that there are enough sample pixels to provide a good sampling for the underlying distribution. This is a reasonable assumption given that skin-color pixels of any particular person are closely clustered in HSV color space [11]. In addition, the recursive nature of the adaptive histogram algorithm requires use of samples from more than one frame; thereby increasing the number of samples used in estimating the distribution at any time.

5.3 Motion of Distributions

As mentioned earlier, skin-color distributions tend to evolve over the sequence of observed frames. In order to model and predict this evolution, we need to make some assumptions about the types of motions that distributions can undergo in the color space.

One assumption is that skin-color distribution evolves as a whole; thus, there cannot be any local deformations or evolutions in the distribution. Furthermore, global deformations of the distribution are assumed to be affine. These decisions are based on observations made in goodness of fit studies [11]. To further simplify our prediction model we constrain ourselves to the three most significant affine transformations: translation, rotation and scaling. We employ an eight-parameter vector defined as follows:

$$\xi = T_1, T_2, T_3, S_1, S_2, S_3, \theta, \phi \tag{3}$$

where T_i are differential translation, S_i differential scaling, and θ and ϕ are differential angles of spherical rotation applied about the mean of the skin-color distribution.

5.4 Estimating Distribution Motion Parameters

Translation parameters T_t at time t can be extracted directly from the difference in means of the HSV skin-color distribution histogram from frame $t-1$ to t.

Scaling can be extracted by considering the eigenvalues of the covariance matrix of the skin-color distribution. Eigenvalues represent the relative scaling of the distribution along the principal directions defined by the eigenvectors of the covariance matrix. Differential scaling S_t along these principal axes is the ratio of the corresponding eigenvalues for the two consecutive frames.

It is assumed that the incremental rotation of the distribution is smooth and relatively small. Given two coordinate frames defined by the eigenvectors of the covariance matrices of the skin-color distributions in the two consecutive frames, our problem is reduced to finding two angles in the spherical coordinate space centered at the mean that would align the two coordinate systems. The first angle can be found as follows:

$$\theta = acos(e_{1,t-1} \cdot e_{1,t}), \tag{4}$$

where $e_{1,t-1}$ is the eigenvector corresponding to the largest eigenvalue at time $t-1$, and $e_{1,t}$ is the eigenvector corresponding to the largest eigenvalue at time t. The axis of rotation v_θ is found via the cross product: $v_\theta = e_{1,t-1} \times e_{1,t}$.

This defines the rotation $R(v_\theta, \theta)$ that will align the corresponding axes of greatest variation. This rotation when applied to $e_{2,t-1}$ and $e_{3,t-1}$ will put them in the plane perpendicular to $e_{1,t}$. In order to align the axes defined by $e_{2,t-1}$ and $e_{2,t}$ as well as $e_{3,t-1}$ and $e_{3,t}$ we need to apply a rotation about $e_{1,t}$. The angle of this second rotation is $\phi = acos((R(v_\theta, \theta) \cdot e_{2,t-1}) \cdot e_{2,t})$.

5.5 Distribution Dynamical Model

In order to estimate and predict the skin-color distribution over time we need to formalize a dynamic motion model. It has been shown that affine motion can be fully expressed

in terms of an auto-regressive Markov process [2]. A second order dynamical process handles both oscillatory and arbitrary translational motion. We will now formulate the discrete second-order Markov process that will be used in our system. The formulation follows [2].

First, we define the N-dimensional state vector X, which in our case is an eight-dimensional parameter vector (Eq. 3). The system's second-order dynamics is defined by a stochastic differential equation [2]. The stochastic portion of the dynamics is modeled by zero mean, unit variance N dimensional Brownian motion. For our application, we utilize the discrete-time model:

$$\begin{bmatrix} X_n - \bar{X} \\ X_{n+1} - \bar{X} \end{bmatrix} = \begin{bmatrix} 0 & I \\ A_0 & A_1 \end{bmatrix} \begin{bmatrix} X_{n-1} - \bar{X} \\ X_n - \bar{X} \end{bmatrix} + \begin{bmatrix} 0 \\ Bw_n \end{bmatrix}. \quad (5)$$

The mean vector $\bar{X}$ corresponds to the observed mean displacement in each of the eight affine parameters. The $N \times N$ submatrices A_0 and A_1 govern the deterministic part of the motion model, whereas submatrix B governs the stochastic part. Rearranging terms yields:

$$X_{n+1} = A_0 X_{n-1} + A_1 X_n + (I - A_0 - A_1)\bar{X} + Bw_n. \quad (6)$$

5.6 Learning Parameters for Dynamical Model

An algorithm for learning the parameters of the proposed second-order Markov dynamical model is needed. The parameters to be learned are A_0, A_1, and B. Unfortunately it is impossible to observe B directly; instead we observe $C = BB^T$. We can estimate these parameters using a standard MLE algorithm described in [3]. This algorithm is used with minor modifications as described in [11].

The eight parameters are treated as independent variables, allowing us to estimate the motion model parameters with fewer observation frames than would be required in the fully-coupled eight-dimensional case. In this case, the minimum number of observation frames required for learning is four. However, more robust performance can be achieved by considering more frames. In experiments, best results were achieved with $n = 8$ to 30. For a real-time NTSC video stream, learning takes less than one sec.

5.7 Histogram Adaptation

Adaptive histograms combine predictions and observations. In our system, color histograms are first normalized to obtain estimates of the actual probability density functions of the skin and background distributions at hand. Updates to histogram bins are made via the following model:

$$H_{i,j,k}(t) = (1 - a)H_{i,j,k}(t - 1) + (a)H^{(p)}_{i,j,k} \quad (7)$$

where i, j, and k designate the bin under consideration and a is a scalar between 0 and 1 that allows us to adjust the speed of adaptation. The histogram $H^{(p)}$ is predicted by the second-order Markov model as described above. Optimal values of the adaptation parameter a can be determined empirically, as discussed in Sec. 7.

6 Prediction and Tracking

The prediction-tracking phase is an extension of the learning phase with one additional construct: the prediction module. This module predicts the future deformations that the distribution will undergo, and hence makes it possible to segment the future frame with a more accurate estimate.

The predicted changes in the translation, rotation, and scaling of the distribution are propagated by warping all color vectors making up the histogram distribution, and then re-sampling it. The new re-sampled distribution is then used to segment the next frame, instead of the previous observation as was done in the learning phase of the system. The rest of the system performs same as before.

6.1 Evolution of Dynamical Model

It is reasonable to assume that not only can a distribution evolve over time, but in addition the process that guides the evolution may change also. This is especially true for long sequences where various illumination changes are expected. In order to handle this, we re-train the motion model as new data becomes available. We always use the last n frames to learn the motion model, hence at any given time t the model will be extracted from $(t-n-2 \ldots t-2)$ frames inclusively. Frames $t-1$ and t define the parameter state vector, and are used to predict the future parameters.

7 Finding Optimal Adaptation Coefficients

As described in Eq. 7, each adaptive histogram has a single adaptation parameter $a = [0, 1]$ that controls the adaptation speed. An adaptation coefficient of $a = 0$ corresponds to a fully in-adaptive histogram, whereas $a = 1$ yields a memoryless histogram representation that is fully-adaptive. Since we have two histograms that we use for two corresponding classes, there are two adaptation parameters that have to be estimated, a_{fg} and a_{bg}, for our system. These parameters can be determined empirically, as is demonstrated in the following example.

We proceed with establishing the optimal foreground adaptation by fixing the background at $a_{bg} = 0$ and varying a_{fg} over its entire effective range while recording the results of segmentation on each of the three 75 frame learning sequences. The resulting segmentation is then compared with the hand labeled ground truth data in order to evaluate the performance. Performance is evaluated using two criteria: the determinant of the confusion matrix and a receiver operator characteristic (ROC) curve.

Fig. 2 shows the result of the experiment described. The adaptation coefficient a_{fg} varies between 0 and 1 by a constant delta of 0.05. The first graph shows the determinant of the confusion matrix as a_{fg} varies. As can be seen in the graph, there is a clear peak that occurs at $a_{fg} = 0.8$. The second graph shows the effects of changing the foreground adaptation coefficient on the ROC curve. The choice of $a_{fg} = 0.8$ is confirmed by the ROC curve.

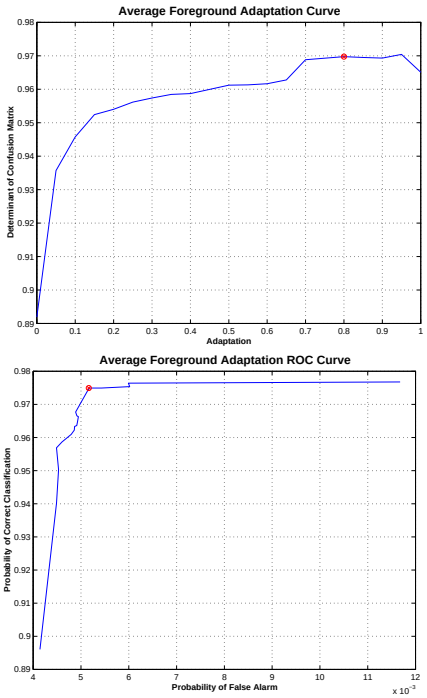

Figure 2: Performance as a function of the foreground histogram adaptation factor a_{fg}. The top graph plots the determinant of the confusion matrix. The bottom graph shows the ROC curve.

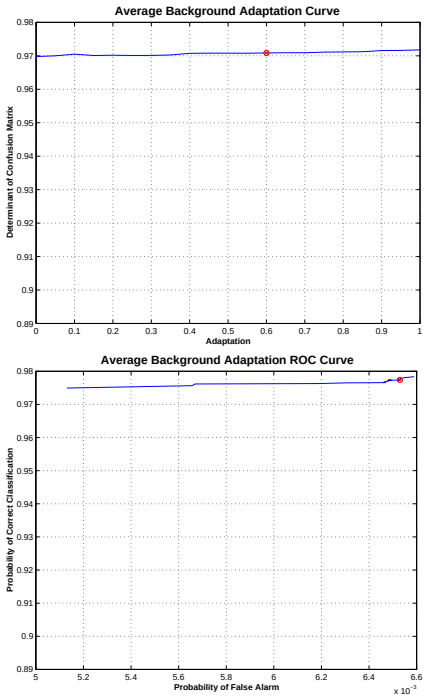

Figure 3: Performance as a function of the background histogram adaptation factor a_{bg}. The top graph plots the determinant of the confusion matrix. The bottom graph shows the ROC curve.

In order to find the optimal adaptation for background we fix the $a_{fg} = 0.8$ and repeat the procedure varying the values for a_{bg}. Fig. 3 shows the two performance curves that were constructed to evaluate the performance of the system at each of the tested values for a_{bg}. The graphs are essentially flat. This can be explained in terms of the training set, which consists of sequences with only very slowly moving background. In general, however we want to be able to handle faster varying backgrounds, and hence we pick a reasonable adaptation value of $a_{bg} = 0.60$.

Two observations arise from this empirical study. First, adaptation of the foreground is more significant than that of the background, which agrees with intuition. The person in front of the camera usually moves much faster than the background; thus, the foreground tends to experience a much greater variation in its color distribution changes, and hence requires a more adaptive model. Second, even though segmentation using adaptive histograms performs better than the static segmentation ($a_{fg} = 0$), the fully-adaptive ($a_{fg} = 1$) setup is not ideal. One reason for this is noise that is present in the segmentation process as well as in the input. The semi-adaptive system suggested by the empirical study ($a_{fg} = 0.8, a_{bg} = 0.60$) tends to be more robust.

8 Experiments

To evaluate the performance of our system we collected a set of 21 video sequences from nine popular DVD movies.[1] The sequences were chosen to span a wide range of environmental conditions. People of different ethnicity and various skin tones are represented. Some scenes contain multiple people and/or multiple visible body parts. Collected sequences contain scenes shot both indoors and outdoors, with static and moving camera. The lighting varies from natural light to directional stage lighting. Some sequences contain shadows and minor occlusions. Collected sequences vary in length from 50 to 350 frames; most, however, are in the 70 to 100 frame range. Fig. 4 shows example frames from the collected sequences.

All experimental sequences were hand-labeled to provide the ground truth data for algorithm performance verification. Every fifth frame of the sequences was labeled. For each labeled frame, the human operator created one binary image mask for skin regions and one for non-skin regions (background). Boundaries between skin regions and background, as well as regions that had no clearly distinguishable membership in either class were not included in the masks and are considered *don't care* regions. The segmentation of these regions was not counted during the experimentation or evaluation of the system. Fig. 5 shows one example frame and its ground-truth labeling.

[1] Test sequences, results and labeled ground-truth are available from the web site: http://www.cs.bu.edu/groups/ivc/ColorTracking/.

Figure 4: Examples frames from sequences used for experimentation.

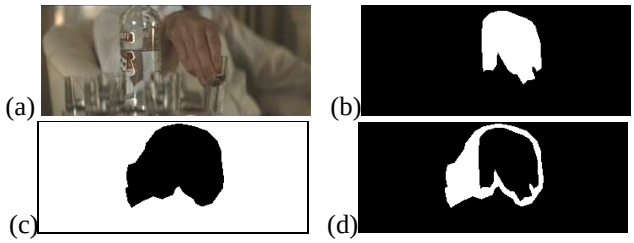

Figure 5: Example of a labeled ground truth frame: (a) original image from a sequence in which a hand is shown reaching to lift a drinking glass, (b) corresponding labeled ground truth mask image for skin, (c) background, and (d) *don't care* regions. Boundaries between skin regions and background, as well as regions that had no clearly distinguishable membership in either class were not included in the masks and are considered *don't care* regions.

8.1 Performance Experiments

The performance of the system was evaluated using the determinant of the confusion matrix criterion. The determinant of the confusion matrix was computed for every hand-labeled frame of the sequence. To gain an aggregate performance metric for the sequence, the average determinant of the confusion matrix was computed.

For comparison, we measured the classification performance of a standard static histogram segmentation approach [6] on the same data set. The static histogram approach implemented used the same prior histograms and threshold as our adaptive system (see Sec. 4.2). The same binary image processing operations of connected component analysis, size filtering, and hole filtering were performed to achieve a fair comparison.

The performance results are outlined in Table 1. Three performance measures were computed: correct classification of skin pixels, correct classification of background pixels, and the determinant of the confusion matrix $Det[C]$. With respect to the $Det[C]$ measure, out of 21 sequences considered, 16 performed better using our dynamical approach. An increase in performance of up to 25% was observed. Performance increase of over 10% was observed on five sequences. Skin classification rates with dynamic histograms were as good or better than the static histogram approach in all cases.

		Classification Performance					
Sequence Info		*Static*			*Dynamic*		
#	# frames	*skin*	*bg*	$Det[C]$	*skin*	*bg*	$Det[C]$
1	71	70.2	97.5	0.67	72.2	96.9	0.69
2	349	64.3	100	0.64	74.8	100	0.75
3	52	92.9	98.5	0.91	96.4	97.8	0.94
4	99	46.2	100	0.46	56.7	99.9	0.57
5	71	90.2	100	0.90	96.9	100	0.97
6	71	96.3	100	0.96	97.5	100	0.98
7	74	90.7	95.4	0.86	91.6	94.0	0.86
8	119	15.1	100	0.15	38.3	100	0.38
9	71	85.9	99.5	0.85	89.8	99.5	0.89
10	71	77.1	91.6	0.69	77.8	89.8	0.68
11	109	92.4	99.7	0.92	94.5	99.5	0.94
12	49	43.1	100	0.43	69.2	100	0.69
13	74	96.9	99.9	0.97	97.6	99.9	0.97
14	74	97.8	100	0.98	98.3	100	0.98
15	90	87.3	100	0.87	86.5	100	0.87
16	75	74.7	100	0.75	84.3	100	0.84
17	72	98.6	98.8	0.97	98.6	98.8	0.97
18	71	81.5	99.8	0.81	88.0	100	0.88
19	71	36.3	100	0.36	37.6	100	0.38
20	71	93.2	37.5	0.31	97.1	36.6	0.34
21	232	83.6	100	0.84	83.4	100	0.83
Average		76.9	96.1	0.73	82.2	95.8	0.78

Table 1: Table of performance figures for the 21 different video sequences from popular DVD movies. The experiments compared classification accuracy for the dynamic *vs.* static histogram approach. Three performance measures were computed: correct classification of skin pixels, correct classification of background pixels, and the determinant of the confusion matrix $Det[C]$.

The five sequences that failed to perform better, had an insignificant performance loss. In all five failure cases, the system performed no worse than 1%. This performance degradation was due to skin-like color patches appearing in

the background of initial frames of a sequence. Recall that these initial frames are used in in estimating the parameters of the Markov model (Sec. 5).

Finally, we performed a set of experiments to establish system stability over time. For example, the graph in Fig. 6 shows system performance on the longest sequence in our test set (349 frames).

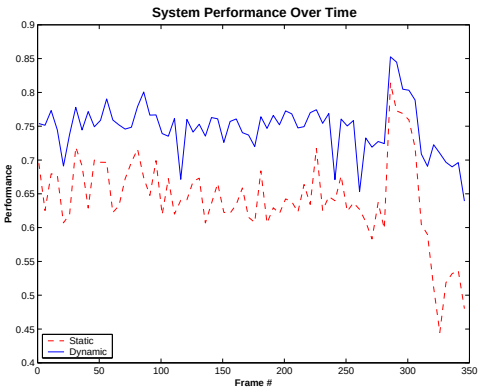

Figure 6: Performance of the dynamical system over an extended sequence. The horizontal axis represents time, measured in frames. The vertical axis represents the performance measured by the determinant of the confusion matrix. The dotted line corresponds to the performance of the static histogram segmentation, and the solid line to our dynamic approach.

As can be seen from the graph in Fig. 6, the dynamic approach was consistently better than the static method in classifying skin and background pixels. Not only does our system perform over 10% better for the entire sequence, it is also more stable. The standard deviation of performance for our system was measured to be 0.0375, which is almost a half of the standard deviation of 0.0630 measured for the static segmentation approach. It should be noted that the stability of our system was consistent across experiments.

In all of the above experiments, adaptation coefficients $a_{fg} = 0.8$ and $a_{bg} = 0.60$ were determined once off-line for a given training set as described in Sec. 7. The adaptation coefficients remain fixed across all trials.

9 Discussion

As exhibited in the experiments, the proposed algorithm generally performs better than the competing stationary histogram approach. The main advantage of the new technique is its stability to fairly rapid changes in the apparent skin-color due to illumination changes, surface interreflection, and rapid changes in background.

In general we noticed that the final result of our algorithm depends greatly on the initialization phase. If the algorithm is initialized with an over-segmented region it generally performs much worse than if it is initialized with an under-segmented version of the same image. This is due to the way adaptation works. In general adaptation facilitates bounded region growing. Initialization and subsequent segmentation accuracy could be further improved via the use of shape and blob-based motion constraints [7], and/or domain-specific constraints like face detection [9].

Furthermore, in our experiments the foreground adaptation had a much higher impact on the final system performance, as opposed to the background adaptation. This was true even for sequences with slowly varying backgrounds. It has been observed that for many sequences one can get away with a very inadaptive background distribution, while maintaining almost the same error rates, as long as foreground adaptation stays the same.

Scene changes are not explicitly modeled by our system; however the system can account for slowly changing dynamic scenes due to the nature of the algorithm. As a possible future extension to the system we are considering automatic re-initialization based on the threshold for the magnitude of change in the background and foreground distributions. It is expected that this would make the system more robust to abrupt scene and illuminant changes.

Acknowledgments

This work was supported in part through ONR Young Investigator Award N00014-96-1-0661, and NSF grants IIS-9624168 and EIA-9623865.

References

[1] S.T. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *CVPR*, 1998.

[2] A. Blake. *Active Contours*. Cambridge U. Press, 1998.

[3] A. Blake, M. Isard, and D. Reynard. Learning to track the visual-motion of contours. *AI*, 78(1-2):179–212, 1995.

[4] T. Darrell, G.G. Gordon, M. Harville, and J. Woodfill. Integrated person tracking using stereo, color, and pattern detection. In *CVPR*, 1998.

[5] W. Hafner and O. Munkelt. Using color for detecting persons in image sequences. *Pattern Rec. and Image Anal.*, 7(1):47–52, 1997.

[6] M.J. Jones and J.M. Rehg. Statistical color models with application to skin detection. In *CVPR*, 1999.

[7] N. Oliver, A.P. Pentland, and F. Berard. Lafter: Lips and face real time tracker. In *CVPR*, 1997.

[8] Y. Raja, S.J. McKenna, and S. Gong. Tracking and segmenting people in varying lighting conditions using colour. In *AFG*, 1998.

[9] S. Satoh, Y. Nakamura, and T. Kanade. Name-it: Naming and detecting faces in news. *IEEE MultiMedia*, 1999.

[10] S. Singh and N. Papanikolopoulos. Vision-based detection of driver fatigue. C.S. TR, U. Minn., 1997.

[11] the authors. . Tech report, 1999.

[12] C.R. Wren, A. Azarbayejani, T.J. Darrell, and A.P. Pentland. Pfinder: Real-time tracking of the human body. *PAMI*, 19(7):780–785, 1997.

[13] J. Yang, L. Weier, and A. Waibel. Skin-color modeling and adaptation. TR *CMU-CS-97-146*, 1997.

[14] M.H. Yang and N. Ahuja. Detecting human faces in color images. TR Beckman Inst., UIUC, 1998.