SKIN DETECTION: A RANDOM FOREST APPROACH

Rehanullah Khan[1], *Allan Hanbury*[2], *Julian Stoettinger*[1]

[1] TU-Wien, Austria
[rehan,julian]@prip.tuwien.ac.at
[2] IRF, Vienna, Austria
a.hanbury@ir-facility.org

ABSTRACT

Skin detection is used in applications ranging from face detection, tracking body parts and hand gesture analysis, to retrieval and blocking objectionable content. For robust skin segmentation and detection, we investigate color classification based on *random forest*. A random forest is a statistical framework with a very high generalization accuracy and quick training times. The random forest approach is used with the IHLS color space for raw pixel based skin detection. We evaluate random forest based skin detection and compare it to Bayesian network, Multilayer Perceptron, SVM, AdaBoost, Naive Bayes and RBF network. Results on a database of 8991 images with manually annotated pixel-level ground truth show that with the IHLS color space, the random forest approach outperforms other approaches. We also show the effect of increasing the number of trees grown for random forest. With fewer trees we get faster training times and with 10 trees we get the highest F-score.

1. INTRODUCTION

Skin detection has a wide range of applications both in human computer interaction and content based analysis. Applications such as: detecting and tracking of human body parts [1], face detection [2], naked people detection, people retrieval in multimedia databases [3] and blocking objectionable content [4], all benefit from skin detection.

The most attractive properties of color based skin detection are: potentially high processing speed, invariance against rotation, partial occlusion and pose change. However, standard skin color detection techniques are affected by changing lighting conditions, complex backgrounds and surfaces having skin-like colors.

The approaches to classify skin in images can be grouped into three types: parametric, non-parametric and explicit skin cluster definition methods. The parametric models use a Gaussian color distribution whereas non-parametric methods estimate the skin-color from the histogram obtained from training data [5].

Skin clustering explicitly defines the boundaries of skin in a given color space, generally termed static skin filters. The main drawback of skin clustering is a high number of false detections. Khan et al. [6] addressed this problem by adapting the skin-color model according to reliably detected faces. On large databases the adaptive approach makes large scale video classification feasible [4].

Color is a low level feature that is computationally inexpensive [7]. Perceptually uniform color spaces like the CIELAB and CIELUV are used for skin detection e.g. by [2]. Orthogonal color spaces like YCbCr, YCgCr, YIQ, YUV, YES try to form as independent components as possible.

Neural networks [8], Bayesian Networks e.g. [9], Gaussian classifiers e.g. [5], and self organizing maps [7] have been used to try to increase the classification accuracy.

Random forest is an ensemble classifier having a quick training phase and a very high generalization accuracy [10, 11, 12]. It is successfully used in image classification [13], image matching [14], segmentation [15] and gesture recognition [16].

We use random forest approach for skin segmentation. We model the skin classification problem in the IHLS color space [17]. We find that the random forest approach has high accuracy for raw pixel based skin segmentation. We evaluate the random forest approach in the IHLS color space and compare it to Bayesian network, Multilayer Perceptron, SVM, AdaBoost, Naive Bayes, and RBF network. Results on a database of 8991 images with manually annotated pixel-level ground truth show that in the IHLS color space, random forest approach outperforms other approaches.

Section 2, explains random forest and the generalization error. Experimental details and the data set used are given in Section 3. Section 4 concludes.

2. RANDOM FOREST

The popularity of tree classifiers is their intuitive appeal and easy training procedures. However there is no classical decision tree approach to increase both classification and generalization accuracy. For this purpose the random forest was introduced by Tin Ho [18]: Random forest is an ensemble of

tree predictors such that each tree depends on the values of a random vector. This vector is sampled independently of the same distribution for all the numerous trees in the forest [10]. To classify a new object from an input vector, the input vector is presented to each of the trees in the forest. Each tree gives a classification, and we say the tree "votes" for that class. The forest chooses the classification having the most votes.

For growing trees, if the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree. If there are M input variables, a number $m << M$ is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing. Each tree is grown to the largest extent possible. There is no pruning. For classification, the final selection by the forest is based on the maximum voting among the trees.

Breiman [10] presents generalization error in terms of the strength of each random tree and the correlation between them. In case of random forest for the kth tree a random vector Θ_k is generated. The independence property enforce that the random vector Θ_k is independent of the past random vectors $\Theta_1, \ldots \Theta_{k-1}$ but with the same distribution. For aggregate tree classifiers $\{h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_k(\mathbf{x})\}$ the margin function can be defined as:

$$mg(\mathbf{X}, Y) = \bar{x}_k I(h_k(\mathbf{X}) = Y) - \max_{j \neq Y}(\bar{x}_k I(h_k(\mathbf{X}) = J))) \quad (1)$$

where $\bar{x}$ is the average and I the indicator function. The generalization error using the margin function is given by,

$$PE^* = P_{\mathbf{X},Y}(mg(\mathbf{X}, Y) < 0 \quad (2)$$

Which indicates the probability over the space of the input vectors $\mathbf{X}$ and class labels Y. Breiman defines the upper bound on the generalization error as,

$$PE^* \leq \bar{\rho}\frac{(1 - s^2)}{s^2} \quad (3)$$

Where $\bar{\rho}$ is the mean correlation between classifiers and s being the strength of the ensemble. Thus generalization error depends on correlation between the random trees and the strength of individual classifiers in the forest. With the increase in tree count the generalization error converges to a limit. In practice as few as 10 trees present competitive results [10]. We evaluate this concept in the following section.

3. EXPERIMENTS

We have used images extracted from 25 videos provided by an Internet service provider that requires a skin detection application for their on-line platform. The sequences contain scenes with multiple people and/or multiple visible body parts and scene shots both indoors and outdoors, with steady or moving camera and varying illumination. The data set is available on-line[1].

A total of 8991 images with annotated pixel-level ground truth are used. Performance is measured using F-score, estimated by equally weighting precision and recall. Evaluation is done for the random forest, Bayesian network, Multilayer Perceptron, SVM, AdaBoost, Naive Bayes and RBF network in the IHLS color space, see Figure 1.

Random forest shows the best performance in terms of accuracy, precision and recall as shown in Table 1. For random forest we tested the effect of growing more than one tree (shown in Figure 2). Even with only one tree (and therefore very fast training times), we reach F-score of 0.672, which is still outperforming all the other approaches. We get the maximum F-score of 0.739 with 10 trees. Addition of more than 10 trees does not increase F-score but rather converges to a stable performance.

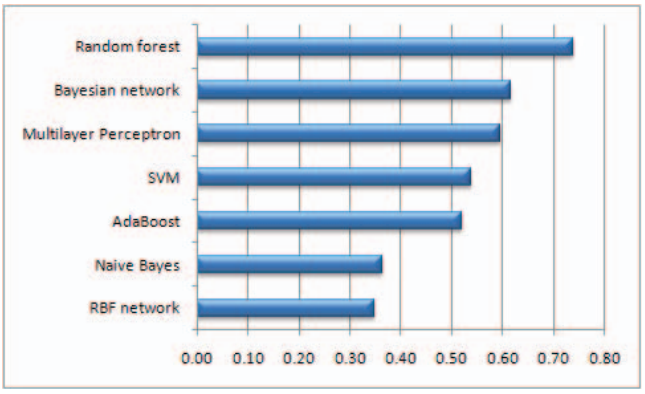


Fig. 1. F-score of different classifiers.

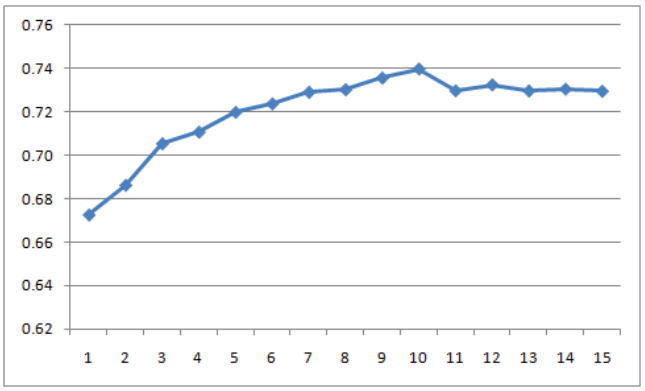
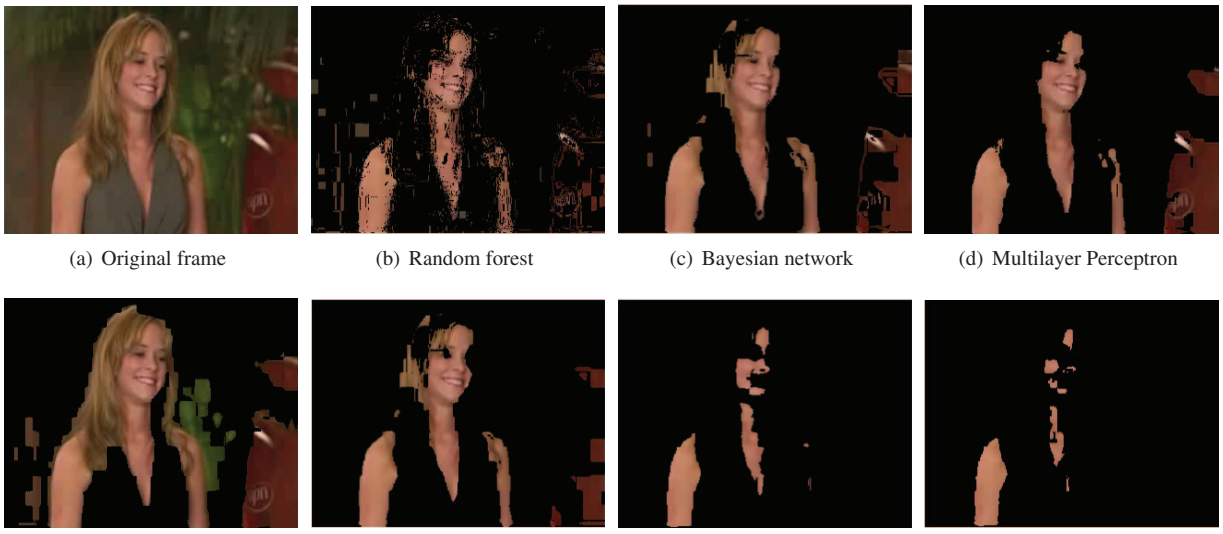


Fig. 2. F-score vs number of trees grown for random forest approach (X-axis: number of trees used. Y-axis: F-score)

[1] http://www.feeval.org



(a) Original frame (b) Random forest (c) Bayesian network (d) Multilayer Perceptron

(e) SVM (f) AdaBoost (g) Naive Bayes (h) RBF network

Fig. 3. Skin detection using different classifiers. Non-skin is black. Detected skin regions are from the original frame.

A brief summary of the classifiers being compared along with the Weka [19] parameters is given below.

Bayesian network: A Bayesian network is a representation for random variables and a conditional independences with in these random variables. For a Bayesian network a Markov boundary of each node is defined. A Markov boundary is related to the independences relationship of nodes. For skin segmentation the parameter settings are: For conditional probability tables of the Bayes network, a simple estimator with Alpha of 0.5 is used, the searching is based on K2 with the number of parents equals 1 and the score type of Bayes. No ADTree is used. As shown in Figure 3(c) and Table 1 the Bayesian approach gives accurate skin model.

Multilayer Perceptron: A multilayer Perceptron is a feedforward artificial neural network model. It maps input data onto a set of appropriate output. Compared to standard linear perceptron, it can distinguish data that is not linearly separable by using a nonlinear activation function. The learning rate selected is 0.3, momentum of 0.2, validation set and seed is null, validation threshold being set to 20. Sample skin detection is shown in Figure 3(d). The accuracy, precision and recall are reported in Table 1.

SVM: SVM searches for the optimal separating hyperplane for inter-class separation. The polynomial kernel is used. The complexity parameter selected is 1, tolerance parameter being 0.0010, epsilon for round off errors of 10^{-12}, the random seed being 1. The visual appearance of SVM in Figure 3(e) reveals more non-skin pixels flagged as skin. The main problem in the model seems to be the non-uniform distribution of the range of the color space components. Color spaces with angular terms suffer with the classification using SVM.

Table 1. Skin segmentation: classifiers statistics.

Classifier	Accuracy	Precision	Recall
Random forest	**0.877**	**0.738**	**0.740**
Bayesian network	0.806	0.588	0.643
Multilayer Perceptron	0.816	0.627	0.565
SVM	0.780	0.545	0.531
AdaBoost	0.798	0.607	0.453
Naive Bayes	0.770	0.545	0.271
RBF network	0.773	0.563	0.250

AdaBoost: AdaBoost is adaptive i.e. subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost is less sensitive to the overfitting problem than most learning based algorithms. For parameter settings: The weight threshold being set to 100, seed is 1, the number of iterations selected are 10 and *Decision Stump* is selected as the base classifier to be used. As shown in Figure 3(f) and Table 1, the weighting of the adaboost gives a reliable and precise model with comparably low recall rates.

Naive Bayes: The Naive Bayes classifier is a specification of Bayes inference with a naive assumption of independence. It is a simple probabilistic classifier providing maximum a posteriori probability for each testing instance. For skin problem, the kernel estimator and the supervised discretization is set to false. Sample skin detection is shown in Figure 3(g). Comparably low accuracy, precision and recall are shown in Table 1.

RBF network: RBF network is a special kind of neural network and uses radial basis functions as activation func-

tions. RBF network generally consists of three layers: an input layer, a hidden layer with a non-linear RBF activation function and a linear output layer. For parameter settings: The clustering seed is set to 1, the number of clusters are set to 2, the ridge value for the logistic/linear regression is 10^{-8}, maximum iterations are set to -1 and 0.1 is selected as the minimum standard deviation for the clusters. For sample skin detection see Figure 3(h). The lowest F-score among all the classifiers can be seen in Figure 1. The parameter tuning is crucial for the classifier. Still, for this noisy data it was not possible to achieve state of the art performance.

Throughout all the experiments, we are using the IHLS color model which is improved to other angular color spaces (HLS, HSI, HSV, etc.) by removing the normalization of the saturation by the brightness [17]. This property overcomes certain numerical problems on the limits of the color channels giving a better distribution in our feature space. For all the classifiers, the three raw components of IHLS i.e. H, L and S are used as the feature vectors. Ten fold cross-validation is used for evaluation. On the data set the random forest has a higher F-score than all other techniques (shown in Figure 1). The random forest approach provides an increased classification performance of almost 12% to Bayesian network, 14% to Multilayer Perceptron, 20% to SVM, 22% to AdaBoost, 37% to Naive Bayes and 39% to RBF network approach.

4. CONCLUSION

Skin color features are located in a simple feature space with a high amount of noise. We analyze pixel based skin classification using different state of the art classifiers on a large per pixel annotated publicly available data set. Due to its explorative nature and hierarchical structure, the random forest generalizes well and robust, outperforming all other approaches significantly. We are able to show that when aiming for faster training times the number of trees grown can be reduced while still maintaining state of the art performance.

5. REFERENCES

[1] Antonis A. Argyros and Manolis I.A. Lourakis, "Real-time tracking of multiple skin-colored objects with a possibly moving camera," in *ECCV*, 2004, pp. 368–379.

[2] J. Cai and A. Goshtasby, "Detecting human faces in color images," *Image and Vision Computing*, vol. 18, pp. 63–75, 1999.

[3] Liang-Liang Cao, Xue-Long Li, Neng-Hai Yu, and Zheng-Kai Liu, "Naked people retrieval based on adaboost learning," in *MLC*, 2002, pp. 1133–1138.

[4] Julian Stöttinger, Allan Hanbury, Christian Liensberger, and Rehanullah Khan, "Skin paths for contextual flagging adult videos," in *ISVC*, 2009, pp. 303–314.

[5] Michael J. Jones and James M. Rehg, "Statistical color models with application to skin detection.," *IJCV*, vol. 46, no. 1, pp. 81–96, 2002.

[6] Rehanullah Khan, Julian Stöttinger, and Martin Kampel, "An adaptive multiple model approach for fast content-based skin detection in on-line videos," in *ACM MM, AREA workshop*, 2008.

[7] D. Brown, I. Craw, and J. Lewthwaite, "A SOM based approach to skin detection with application in real time systems," in *BMVC'01*, 2001, pp. 491–500.

[8] Jae-Young Lee and Yoo Suk-in, "An elliptical boundary model for skin color detection," in *ISST*, 2002, pp. 579–584.

[9] Nicu Sebe, Ira Cohen, Thomas S. Huang, and Theo Gevers, "Skin detection: A Bayesian network approach," in *ICPR*, 2004, pp. 903–906.

[10] Leo Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[11] Raphael Maree, Pierre Geurts, Justus Piater, and Louis Wehenkel, "Decision trees and random subwindows for object recognition," in *ICML*, 2005.

[12] Pierre Geurts, "Extremely randomized trees," in *ML*, 2003.

[13] Raphael Maree, Pierre Geurts, Justus Piater, and Louis Wehenkel, "A generic approach for image classification based on decision tree ensembles and local subwindows," in *ACCV*, 2004, pp. 860–865.

[14] Vincent Lepetit, Pascal Lagger, and Pascal Fua, "Randomized trees for real-time keypoint recognition," in *CVPR*, 2005, pp. 775–781.

[15] Pei Yin, Antonio Criminisi, John Winn, and Irfan Essa, "Tree-based classifiers for bilayer video segmentation," in *CVPR*, 2007.

[16] T. Deselaers, A. Criminisi, J. Winn, and A. Agarwal, "Incorporating on-demand stereo for real time recognition," in *CVPR*, 2007.

[17] Allan Hanbury, "A 3d-polar coordinate colour representation well adapted to image analysis," in *SCIA*, 2003, pp. 804–811.

[18] Tin Kam Ho, "Random decision forests," in *ICDAR*, 1995, pp. 278–282.

[19] G. Holmes, A. Donkin, and I. H. Witten, ," in *Intelligent Information Systems*, 1994, pp. 357–361.