Real-time Face Detection Comparison Analysis

Haisheng Wu and John S. Zelek
Robot Vision Group
University of Guelph
Guelph, ON. N1G 2W1

Abstract
Two real-time face detection techniques are compared. From analyzing the theory of the techniques and empirical results, a three part strategy has been observed to be necessary for designing real-time face detection systems which also achieve high detection rates. The three critical factors of real-time face detection are using Wavelets to represent features, machine learning approaches for pattern classification and finally using a smart searching strategy to scan the windows. Future improvements include using a preliminary stage for skin classification in order to localize the computation and improve performance time.
Keywords: Real-time, Face detection, Wavelets, HSV model, Neural Network, Statistic Model.

1 Introduction

Face detection has become an active research area and has progressed since the early 1990's. Real-time performance alongside robustness are essential ingredients but are still challenging. The main challenge in face detection is the amount of variation in visual appearance, such as shape, size, color, surrounding environment, light condition, shadows, pose (including position and orientation).

A general statement of this problem could be formulated as follows: Given video images of an unconstrained scene, determine the location and size of each human face (if any) within a short time, such as processing 25 frames per second for PAL, or 30 frames per second for NTSC. The solution to the problem involves segmentation, extraction, and verification of faces from an unconstrained background, irregardless of illumination, orientation, and camera distance.

Real-time face detection is important because it is usually the first step of a fully automatic human face recognizer. In particular, a real-time face detector can have broad practical applications, such as robot vision, user interfaces, image database, teleconferencing, image indexation engine, surveillance and census systems, etc.

There are two main categories of real-time face detection techniques in accordance with their prior face knowledge. The feature-based algorithm uses face knowledge and follows the classical detection methodology in which low level features are derived prior to knowledge-based analysis. The typical properties of the face such as motion, skin color, face geometry are exploited by manipulating distance, edge, angles, and area measurements of the visual features derived from the scene. The other approach is image-based and takes advantage of the current advances in pattern recognition theory. Image-based representations of faces are directly classified into a face group using mapping and training schemes.

1.1 Feature-based Approaches

Feature-based approaches are roughly classified as three main groups: local feature analysis, global feature analysis and active shape models.

Local feature analysis involves locating faces by utilizing the facial properties, such as edges, color, motion and others. As the most primitive feature, edge detection is applied to locate facial features. Many different edge-detection-based approaches for face detection have been proposed, such as the Sobel operator, as well as steerable and multi-scale-orientation filters. The low-level nature causes features generated from this analysis to be ambiguous. Color is also a powerful means for face detection. In contrast to RGB, YIQ, YES and YUV color models, the HSI color representation has shown to have some advantages in giving large variance among facial feature color clusters, so it is used to extract facial features such as lips, eyes, and eyebrows (Garcia[4]). Motion information is a convenient means of locating faces if the use of a video sequence is available. The existence of an eye-pair, the estimation of moving images contours, and optical flow are also used for face detection.

Besides local-feature-face-detection, the knowledge of face geometry can be utilized as a global feature to detect faces. Face-shape, eye-pair, outline (top of head) and body (below the head) can also be used to detect faces.

Active shape models range from snakes to the deformable templates and recent Point Distributed Models (PDM) developed for the purpose of complex and non-rigid feature extraction such as eye pupil and lip tracking. Snakes, or active contours, are usually used to locate a head boundary.

Some algorithms rely heavily on a rigid face under constrained conditions and fail detecting faces in unstructured backgrounds. Feature-based methods are applicable for real-time systems where color and motion is available. Since exhaustive multi-resolution window scanning is not always preferable, feature-based methods can provide visual cues to focus attention. In these situations, the most widely used technique is skin color detection based on one of the color models.

1.2 Image-based Approaches

Unlike the feature-based approach, image-based representations of faces are directly classified into face and non-face prototype classes using mapping and training schemes without feature derivation and analysis. This eliminates the potential of modelling error due to incomplete or inaccurate face knowledge. Image-based approaches are roughly categorized into three groups: linear subspace methods, neural networks and statistical approaches.

Linear subspace methods involve principal component analysis (PCA), linear discriminant analysis (LDA), and factor analysis (FA). The PCA technique uses eigenvectors of the covariance matrix of the face distribution to efficiently represent human faces. PCA is an intuitive and appropriate way of constructing a subspace for representing an object class within many classes. However, for modelling the manifold of face images, PCA is not necessarily optimal. Face space might be better represented by dividing it into subclasses, and several methods have been proposed for this, of which most are based on a multi-dimensional Gaussian mixture [10].

Neural networks have become a powerful technique for pattern recognition problems, including face detection. Neural networks today are much more than just simple MLP (Multi Layer Perceptrons). Modular architectures, committee-ensemble classification, complex learning algorithms, auto associative and compression networks, and networks evolved or pruned with genetic algorithms are all examples of the widespread use of neural networks in pattern recognition.

Rowley et al. [7] proposed a retinal connected neural network which incorporates face knowledge. The neural network is designed to look at 20 x 20-pixel windows. One hidden layer with 26 units looks at different regions based on facial feature knowledge. To lower the false positive rate, two networks were applied to tested windows. The dominant factor in the running time of the Rowley system is the number of 20 x 20 pixel windows which the neural network must process. To speed up the implementation, a skin-color detector was proposed to restrict the search region. The pre-processing method was adopted from Sung and Poggio's system [10], including boundary elimination, lightness correction, and histogram equalization.

Raphael et al. [3] proposed a fast face detector based on constrained generative model (CGM). The architecture of the face detector is hierarchical: at each stage a percentage of the hypothesis is excluded. The system consists of four different face filters. These filters, from the simplest, fastest, and less accurate to the most complex, slowest and most accurate, are the following: (1) a motion filter, (2) a skin color filter, (3) a multilayer Perceptron, and (4) a CGM-model filter.

Apart from linear subspace methods and neural networks, there are several other statistical approaches to face detection. Based on an earlier work of maximum likelihood face detection, Colmenarez and Huang [1] proposed a system based on Kullback relative information (Kullback divergence). In Osuna et al. [5], a support vector machine (SVM) is applied to face detection. Schneiderman and Kanade [9] describe two face detectors (frontal view and side view) based on a Bayesian decision rule.

Image-based approaches are the most robust technique for processing gray-scale static images. Almost all of these algorithms are based on multi-resolution window scanning to detect faces at all scales, rendering them computationally expensive. Multi-resolution window scanning can be avoided by combining the image-based approach with a feature-based method as a pre-processor or with the intent of guiding the search based on visual clues such as skin color, including the likelihood of achieving real-time tasks.

This paper compares and analyzes two real-time face detectors which are based on a neural network (Viola & Jones[11]) and statistical information (Schneiderman & Kanade[9]), respectively. Their benefits and shortcomings are described. Based on the analysis of some real-time face detectors, some rules are infered and a new system is proprosed. Section 2 analyzes a real-time neural network face detector using an "integral image" and an adaboosted Perceptron classifier. Section 3 analyzes a statistical face detector which combines a Bayes decision rule with the Wavelets transform. Section 4 describes the implementation issues and section 5 shows the experimental results. Section 6 highlights improvements for better detection results.

2 Theoretical Analysis

2.1 Neural Network Approaches

Viola and Jones proposed a real-time face detector based on a neural networks and a stage cascade. To achieve real-time results with high detection rate, three main techniques - rectangle features, The AdaBoost learning algorithm and the stage cascade are adopted to the classification system. The AdaBoost and stage cascade detector will be discussed later.

2.1.1 Hypotheses

Given a set of features and a training set of face and non-face images, a neural network can be trained to learn a classification function. A very small number of these features can be combined to form an effective classifier.

2.1.2 Rectangular Features

Contrary to most neural network techniques, this system does not work directly with image intensities but simple rectangle features. The key motivation for using simple features rather than pixels is that feature-based systems execute much faster than pixel-based systems.

Rectangle features are derived from Harr basis functions which have been used by Papageorgiou et al. [6]. The Harr filter is utilized to capture vertical and horizontal features, such as edges and bars. In particular, four types of rectangle features (Fig.1) are used. The rectangular feature is the difference between the sum of the pixels in the white region and the sum of pixels in the black region.

In order to compute these features very rapidly at many scales, a texture mapping technique called a Summed-area Table (Fig.1) that is used. The key idea is to replace the texture map with a pre-computed summed-area table texture map. The summed-area table texture map at location x, y contains the sum of pixels above and left of x, y, inclusive:

$$ii(x,y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

where $ii(x,y)$ is the summed-area table texture map and $i(x,y)$ is the original texture map. Using

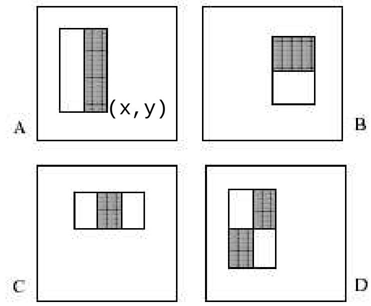

Figure 1: Rectangle features example. The value of the summed-area table texture map at point x, y is the sum of all the pixels above and to the left.

the following pair of recurrences:

$$s(x,y) = s(x, y-1) + i(x,y) \quad (2)$$
$$ii(x,y) = ii(x-1,y) + s(x,y) \quad (3)$$

here $s(x,y)$ is the cumulative row sum, $s(x,-1) = 0$, and $ii(-1,y) = 0$. The summed-area table texture map can be computed in one pass over the original image. Using the summed-area table, a set of over complete rectangular features are computed very quickly.

2.1.3 Neural Network Classifier

After applying the Harr filter, all rectangle features are fed into a neural network classifier, such as the Perceptron. The Perceptron utilizes gradient descent and the E-M algorithm for training. The benefit in using the Perceptron is that it can separate the input pattern into two classes extremely quickly and it only requires moderate storage capacity. But the critical limitation is that Perceptron is only suitable for linearly separable patterns. In order to keep its benefit and overcome its disadvantage, the AdaBoost learning algorithm is applied to boost the classification performance, which will be discussed later.

2.2 Statistical Approaches

Schneiderman and Kanade designed a statistical face detector based on statistical models and probabilistic learning. To construct a statistical model, they adopted the product of multi-dimensional receptive field histograms, which represent the approximation of the probability density function for jointly localized appearance. To capture the jointly localized attributes in space, frequency, orientation and position, 2D Gabor Wavelet transforms are performed to the image. Given a image, the classification function is based on the likelihood ratio test of face and non-face model, which is equivalent to the Bayes decision rule (MAP decision rule).

2.2.1 Statistical Model

Two statistical distribution models, face $P(image \mid face)$ and non-face $P(image \mid non-face)$, account for face objects and the rest of the world. The assumption under this approach is that facial features (eyes,cheeks) occur frequently on the face but occur infrequently in the rest of the visual world. The detection decision is computed using the likelihood ratio test, which is equavilent to the Bayes decision rule:

$$\frac{P(image \mid face)}{P(image \mid non-face)} > \lambda \tag{4}$$

where

$$\left(\lambda = \frac{P(face)}{P(non-face)}\right)$$

If the likelihood ratio in equation(4) is greater than the λ, a face is present. The likelihood ratio test will be optimal if representations for $P(image \mid face)$ and $P(image \mid non-face)$ are accurate.

2.2.2 Multi-dimensional Histograms

The challenge in modelling $P(image \mid face)$ and $P(image \mid non-face)$ is that the true statistical distributions for both faces and the background are unknown. Nobody can show whether face distributions are Gaussian, Poisson, or Multi-modal. The reasonable method is to choose a flexible model accommodating a wide range of distributions. Neural network approaches (MLP or a mixture model) is only a local optimal method, and its gradient descent training is very slow. Multi-dimensional receptive histograms have been used as a practical and reliable real-time means for the approximation of the probability density function [8]. Each histogram, $P_k(pattern \mid object)$, represents the probability of appearance over some specified visual attribute, $pattern_k$. Therefore, face and non-face models are approximated by the product of conditional probabilities of different attributes:

$$P(image \mid face) \approx \prod_k P_k(pattern_k \mid facef) \tag{5}$$

$$P(image \mid non{-}face) \approx \prod_k P_k(pattern_k \mid non{-}face) \tag{6}$$

2.2.3 2D Gabor Wavelets Representation

To represent the face and non-face model, a 5/3 linear phase filter-bank stages based on 2D Gabor Wavelets are applied to capture the jointly localized statistics in space, frequency, orientation and positoin. 10 sub-bands are produced as shown in Fig.2, where LL denotes low frequency, low resolution features of the image, and LH denotes vertical features, such as lines and edges, and HL denotes horizontal features, as well as HH denotes diagonal features. These representations are used as a basis for specifying visual attributes.

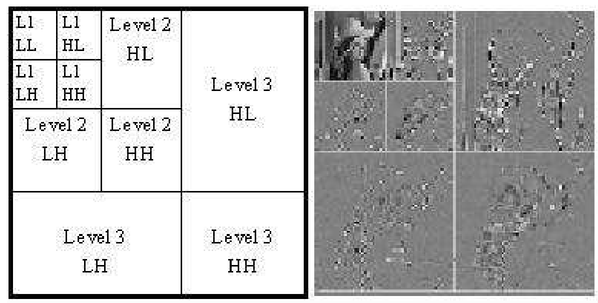

Figure 2: 2D Gabor Wavelets representation of an image

Each visual attribute, pattern (k), represents the values of a subset of quantized Wavelet coefficients. Every attribute is sampled to 8 Wavelet coefficients, each subsequently quantized to 3 levels (Fig.3). This quantization scheme gives $3^8 = 6561$ discrete values for each visual attribute. Overall, 17 attributes are used to sample the 2D Gabor Wavelets into a group of 8 coefficients in four categories ([2]): 1) 6 Intra-sub-band coefficients represent local frequency and orientation visual cues; 2) 4 Inter-frequency coefficients represent visual cues that span a range of frequencies such as edges; 3) 6 Inter-orientation coefficients represent cues that have both horizontal and vertical components such as corners; and 4) 1 Inter-frequency inter-orientation coefficients represent cues that span a range of frequencies and orientations.

2.2.4 Probabilistic Learning

To train face detectors, each histogram is estimated. To estimate each histogram, a count of how often each pattern occurs at each position is initiated in the appropriate set of training examples. To test the image, the 2D Gabor Wavelet coefficients are computed and sampled to find the probability in the lookup table.

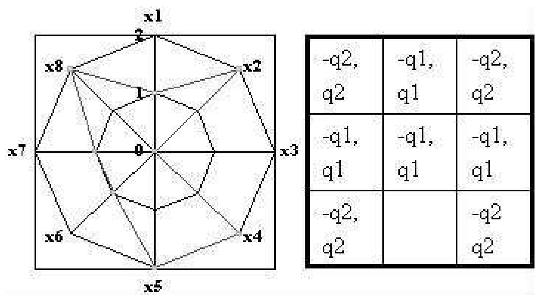

Figure 3: Multi-dimensional histogram representation and Gabor Wavelets coefficient quantization

The final form of the detector is:

$$\frac{\prod_{x,y \in region} \prod_{k=1}^{17} P_k(pattern_k(x,y),x,y \mid face)}{\prod_{x,y \in region} \prod_{k=1}^{17} P_k pattern_k(x,y),x,y \mid non-face)} > \tag{7}$$

where "region" is the scanning window classified.

3 Implementation Issues

3.1 Training Data Set

All face and non-face images were downloaded from the World Wide Web randomly. All images are color and of varying quality. The face and non-face training set are labelled by hand and scaled and aligned to a base resolution of 40x30 pixels. The entire image data set involves 500 pictures which consist of 300 face images and 200 non-face images. In this project, the first 350 images were used for training and the final 150 images for testing.

3.2 AdaBoost Classifiers

Both the AdaBoost learning algorithm and the maximum entropy discrimination technique are used to minimize the classification error. The Adaboost algorithm is used to both select features and boost the classification performance of weak learners by combining a collection of weak classification functions to form a strong classifier. In using the AdaBoost to train the neural network classifier, the final AdaBoosted classifier is a weighted linear combination of T classifiers where the weights are inversely proportional to the training errors (shown on Fig. 4). In using the AdaBoost to train the statistics face detector, each attribute histogram is collected by counting, but each bin is increased based on the weight assigned to the current training examples. All weights are scaled and rounded appropriately.

The benefit of the AdaBoost algorithm is that it does not increase the complexity of the classifier. However it increases the CPU time when using it for neural network classifier. Please note that AdaBoost does not increase the CPU time when used for the statistical classification because the statistical classifier is based on a lookup table technique.

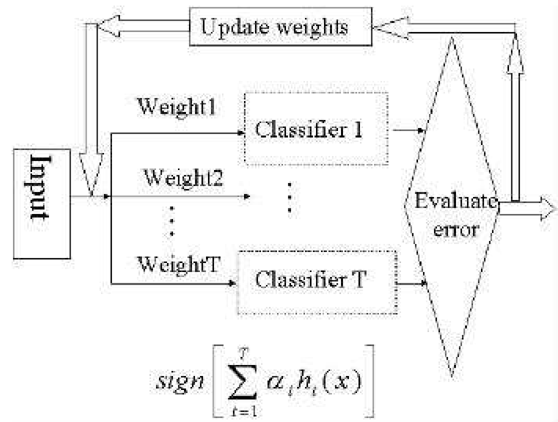

Figure 4: AdaBoosted classifier

3.3 Coarse to Fine Search Strategy

Most image-based face detector exhaustively search the image in position and scale based on a scanning-window. To speed up this algorithm, a heuristic coarse-to-fine strategy is applied in the Schneiderman statistical approach. Firstly, the likelihood ratio for each possible object location is evaluated using low resolution visual attributes, then it is further evaluated at higher resolutions for the object candidates.

3.4 Decision Tree

To reduce computational time, a cascade of classifiers were constructed in the Viola& Jones neural network approach, which is a degenerate decision tree. The critical motivation is that simple and efficient cascade classifiers can reject most of non-face sub-windows while detecting almost all face sub-windows. Viola and Jones proposed a 32 layer cascade of classifiers which involves 4297 features.

4 Experiment Results

Both face detection systems were trained in two stages. Firstly, the Perceptron classifier was trained with image data set using a gradient descent error-convergence algorithm. The training stage took a lot of CPU time to achieve a low error. The experiments demonstrated that the learning may oscillate and could not further decrease the epoch error with a larger learning rate a. Also, the trained Perceptron classifier could only correctly detect the face in the range of 42% and 71%. Obviously, the Perceptron is not suitable for non linear pattern classification, such as face detection, even though its computation is extremely fast.

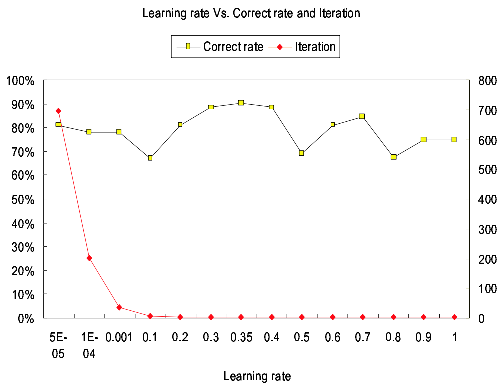

Figure 5: Learning rate VS. correct rate and Iteration on AdaBoost+Perceptron.

The second stage of training is to apply the AdaBoost learning algorithm to the Perceptron result for boosting the performance of Perceptrons. Figure 5 illustrates the AdaBoost learning algorithm remarkably enhancing the performance of the classification functions. The best correct rates of classification reached up to 88% after applying the Adaboost algorithm to the Perceptron result. Figure 5 also demonstrates that there is no significant correlation between correct rate and Perceptron learning rate. Unlike the simple Perceptron, the learning rate can not affect the performance of classification any more when the Perceptron was boosted using AdaBoost algorithm. In other words, the Adaboost learning algorithm decreases the parametric sensitivity of the classification function and makes the face detection system more robust. In addition, we can increase the Perceptron learning rate to dramatically reduce the training time, and keep the correct rate at a high level at the same time. In conclusion, AdaBoosting can reduce CPU time and produce many benefits in training stage.

To demonstrate how T hypotheses (boosting bounds) affect the performance of classification, including correct rate and CPU time, an experiment was simulated and the result is shown in Figure 6. From this figure, we may safely conclude that the CPU time is proporational to the T hypotheses. Additionally, the correct rate improves with an increase in the number of T hypotheses and reaches the highest level at 88% when T is equal to 400. But correct rate will go down if T hypotheses increases more than 400.

Most image-based approaches are based on ex-

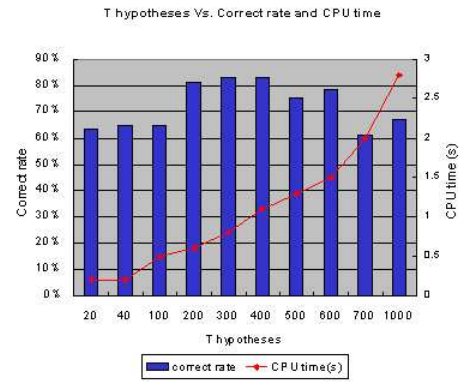

Figure 6: T hypotheses VS. correct rate and CPU time. Bar is T bounds, curve is CPU time.

haustive search and are computationally expensive. To achieve real-time performance, Viola and Jones proposed a decision tree to speed up the window scanning process but they still employed an exhaustive searching method. In this research, we do not repeat this searching approach because it is not suitable for real-time tasks.

For the statistical approach, the first stage of training is to estimate all attribute histograms separately (total of 34 attributes, 17 for face model and 17 for non-face model). To estimate each histogram, each pattern occurance is tabulated at each position in the appropriate set of training examples. The second stage of training using AdaBoost is to increase each bin based on the weight assigned to the current training example. Other than neural network approaches, the AdaBoost learning algorithm does not increase the CPU time in the testing stage since statistical approaches retrieve the probability from a lookup table the in testing stage directly. So, statistical and probabilistic approaches are suitable for real-time tasks.

Algorightm	Detection rate	CPU time
Neural Networks	88.0%	0.0018s
Statistics approach	91.8%	0.0013s

Table 1: Comparison of detection rate and CPU time for window detection

5 Discussions and Conclusions

From the above theoretical analysis and experiments, we can generalize some conclusions in designing real-time face detection systems. Most techniques employ a three part strategy in real-time face detection systems while achieving high detection rate.

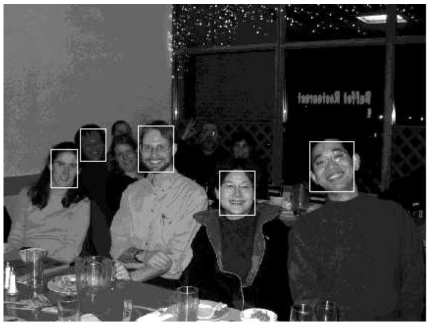

Figure 7: Detection example by the statistical approach.

Firstly, to cope with the variations in visual appearance such as shape, size, color, surrounding environment, light condition, shadows, pose, orientation and position, both techniques use Wavelet transforms to capture the features for learning, where one uses Harr-like Wavelets and the other selects 2D Gabor Wavelets. The key benefit in using Wavelets is that reconstruction of the image is possible, so the image can be classified to face or non-face exactly. The limitation in using Wavelets is that perfect orthogonal Wavelets may increase the computational complexity and consume lots of CPU time since most perfect Wavelets are a non-linear transform (except the Harr Wavelet).

After the first step, both employ machine learning approaches to achieve a high detection rate. The first choice is to construct a classification function based on artificial neural networks using gradient descent error-converge learning, while the other choice is to construct a statistical model based on multi-dimensional histograms using probabilistic learning and approximation for the classification function. To improve the correct rate, both procedures adopted the AdaBoost learning algorithm to decrease the classification error. The advantage in using AdaBoost is that it improves the detection remarkably while it does not increase the computational complexity. The drawback of AdaBoost is that it increase the CPU time in neural networks approaches. However, AdaBoost does not increase the CPU time in statistical approaches since the probabilistic approximation is based on a lookup table. At this point, statistical and probabilistic approaches are suitable for real-time tasks. The main limitation of statistical approaches is that the maximum likelihood ratio heavily depends on the training data set and its selection is only based on experiments.

Finally, a smart-searching strategy is applied to the scanning-window to speed up the face detectors. Since the scanning-window stage along multi-scale, multi-position and multi-orientation is usually time consuming. The best choice to speed up detectors is to adopt some smart-searching strageries. Viola and Jones still employed exhaustive searching methods but they use cascade detectors, or degenarate decision trees to decrease the window processing time. And Schneiderman and Kanade selected a coarse-to-fine strategy based on probabilistic approximation to avoid exhausitive searching stage. In window processing stage, both methods can achieve good results for real-time task. But in scanning-window stage, coarse-to-fine strategy is better than an exhausitive searching stratagy.

6 Future Development

In accordance with the above analysis, our future research will place emphasis on the scanning-window stage to speed up the detector while achieving high detection rate. We are currently investigating constructing a set of filters which are composed of HSV skin color filter, face shape and color homogeneity filter, and a statistical approximation classifier.

6.1 Skin Color Segmentation

A HSV (Hue, Saturation, Value) model can be used to segment color image according to skin color characteristics because sample skin color forms a single and compact cluster in HSV color subspace, which is shown in Fig 8. In contrast to most approaches based on a Bayesian decision rule for skin color classification, six bounding equations in HSV color space are used to segment skin color; which was proposed by Garcia and Tziritas [4]. The HSV color model is performed to find candidate skin regions while avoiding exhaustive search.

$$\begin{cases} S \geq 10, & \\ V \geq 40, & \\ S \leq -H - 0.1V + 110, & \\ H \leq -0.4V + 75, & \\ S \leq 0.08(100 - V)H - 0.5V, & \text{if } H \geq 0 \\ S \leq 0.5H + 35, & \text{if } H < 0 \end{cases}$$

6.2 Skin Region Labelling

Image pixels can be classified as skin pixels or non-skin pixels using the HSV skin color model, but one can not say whether it is face or non-face at the pixel level. Therefore, we need to group skin pixels into different meaningful regions such as a face or a hand etc. A 8-connected component labeling algorithm is applied to form candidate skin areas.

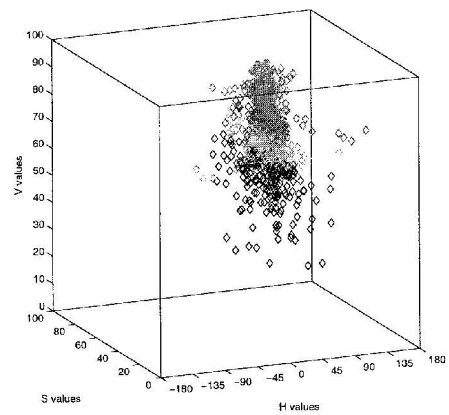

Figure 8: Sample skins color distribution in HSV subspace.

6.3 Shape and Homogeneity Analysis

Since there are many non-face skin regions in the image, a shape and color homogeneity filter is applied to discard most non-face regions.

To analyze face shape quickly, the aspect ratio of the face bounding box is considered because the height to width ratio of a human face falls within a small range, such as the golden ratio ($(1+\sqrt{5})/2 \pm \varepsilon$,where tolerence $\varepsilon = 0.65$). If the aspect ratio of a candidate skin region falls within the golden ratio, it is considered a face region.

Moreover, skin color homogeneity filter is performed by computing the density of skin color pixels in each candidate face region. If the percentage of skin in a candidate region is greater than a threshold ($\tau = 0.6$), the candidate region represents a candidate face region.

6.4 Face Region Classification

In this stage, all candidate face regions are fed into a statistical classifier to determine whether it is a face or not, which is based on a Bayesian decision rule mentioned in section 2.2. The face and non-face model are constucted by the product of multi-dimensional histograms, and a 5/3 linear phase filter-bank is applied to capture the jointly localized statistics in space, frequency, orientation and position.

By combining a skin color filter, shape and homogeneity analysis, with a statistical classifier, we take advantage of the smart-searching approach and overcome the drawback of exhausitive searching method, and approach real-time face detection.

Acknowledgements: The authors would like to thank NSERC (National Science and Research Council) for funding this project.

References

[1] A. J. Colmenarez and T. S. Huang. Maximum likelihood face detection. In *IEEE Proc. of 2nd Int. Conf. on Automatic Face and Guesture Recognition*, pages 222–224, Vermont, 1996.

[2] P.C. Cosman, R.M. Gray, and M. Vetterli. Vector quantization of image subbands: A survey. *IEEE Trans. on Image Processing*, 5(2):202–225, Feb. 1996.

[3] Raphael Feraud, Olivier J. Bernier, Jean-Emmanuel Viallet, and Michel Collobert. A fast and accurate face detector based on neural networks. *IEEE trans. on Pattern Anal. and Machine Intell*, 23(1), January 2001.

[4] C. Garcia and G. Tziritas. Face detection using quantized skin color regions, merging and wavelet packet analysis. *IEEE Trans. Multimedia*, 1:264–277, 1999.

[5] E. Osuna, R. Ferund, and F. Girosi. Training support vector machines: An application to face detection. In *IEEE Proc. of Int. Conf. on Computer Vision and Pattern Recognition*, volume 6, 1997.

[6] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *International Conference on Computer Vision*, 1998.

[7] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. Pattern Anal. Mach. Intell*, 20:23–28, January 1998.

[8] Bernt Schiele. Recognition without correspondence using multidimensional receptive feild histograms. *International Journal of Computer Vision*, 36(1):31–52, 2000.

[9] H. Schneiderman and T. Kanade. A statistical model for 3d object detecion applied to faces and cars. In *IEEE Conf. on Computer Vision and Pattern Recog.*, 2000.

[10] K.-K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 20:39–51, 1998.

[11] Paul Viola and Michael Jones. Robust real-time object detection. In *2nd International Workshop on Statistical and Computational Theories of Vision-Modeling, Learning, Computing, and Sampling*, Vancouver, July 13, 2001.