Robust Hand Detection in Still Video Images Using a Combination of Salient Regions and Color Cues for Interaction with an Intelligent Environment[1]

T. Plötz, J. Richarz, and G. A. Fink

Intelligent Systems Group, Robotics Research Institute, Dortmund University of Technology, Otto-Hahn-Straße 8, 44227 Dortmund, Germany

e-mail: Thomas.Ploetz@udo.edu; Jan.Richarz@udo.edu; Gernot.Fink@udo.edu

Abstract—The "intelligence" of an intelligent environment is not only influenced by the functionality it offers, but also largely by the naturalness and intuitiveness of its interaction modes. A very important natural interaction mode is gestures, as long as the environment's interface poses no strict constraints on how the gestures may be performed. Since gestures are generally defined by hand/arm poses and motions, an important prerequisite to the recognition of unconstrained gestures is the robust detection of hands in video images. However, due to the strongly articulated nature of hands and the challenges given by a realistic (i.e., not strictly controlled) environment, this is a very challenging task, because it means hands need to be found in almost arbitrary configurations and under strongly varying lighting conditions. In this article, we present an approach to hand detection in the context of an intelligent house using a fusion of structural cues and color information. We first describe our detection algorithm using scale-invariant salient region features, combined with an efficient region-based filtering approach to reduce the number of false positives. The results are fused with the output of a skin color classifier. A detailed experimental evaluation on realistic data, including different cue fusing schemes, is presented. By means of an experimental evaluation on a challenging task, we demonstrate that, although each of the two different feature types (image structure and color) has drawbacks, their combination yields promising results for robust hand detection.

DOI: 10.1134/S1054661808030097

1. INTRODUCTION

Recent developments of sophisticated pattern recognition techniques nowadays make even the complex analysis of perception related sensor data manageable. In combination with the meanwhile (almost) ubiquitous availability of powerful computing hardware, intelligent technical systems become possible allowing for more intuitive and, thus, natural human–machine interaction (HMI). It is the focus of such systems to be oriented towards the particular users' needs and desires instead of towards technical limitations.

But what makes humans perceive a related technical system as *intelligent*? An interaction partner is considered smart if humans may interact with him in the way they normally would with other humans, and he shows reasonable reactions to their actions. Thus, an *intelligent* system is not only defined by the services it offers (however useful they may be), but also—and more importantly—by the naturalness of interfaces it offers to access these services. Consequently, human–machine interfaces are sought that resemble natural means of human–human communication.

The key modalities used by humans during interaction in the above mentioned "natural" way are speech and gestures. If a person is able to interact with a particular technical system by talking to it and, especially, articulating supporting gestures, the overall communication will appear more natural and, consequently, much easier to the human. Gestures are usually directly connected to acoustic utterances with regard to their temporal occurrences. Dynamic gestures are frequently used as some kind of "illustration" of the particular utterances or even without any special contextual relations to the particular interaction. The most prominent use of gestures is, however, their application as illustrating pointing signs. In our work we concentrate on the analysis of gestures belonging to the latter type, e.g., for controlling certain components of electrical equipment included in typical households (lights or sunblinds).

Since gestures are mostly defined by hand/arm poses and motions, a fundamental prerequisite to their recognition is the robust detection of hands in images. This paper addresses this fundamental stage of gesture recognition especially focusing on unconstrained gestures recorded in (almost) arbitrary environments.

An obvious approach to hand detection corresponds to the detection of skin-color–like regions within images of a scene. Especially in officelike environments (with furniture which often has skin-like color-

[1] The text was submitted by the authors in English.

Received December 3, 2007

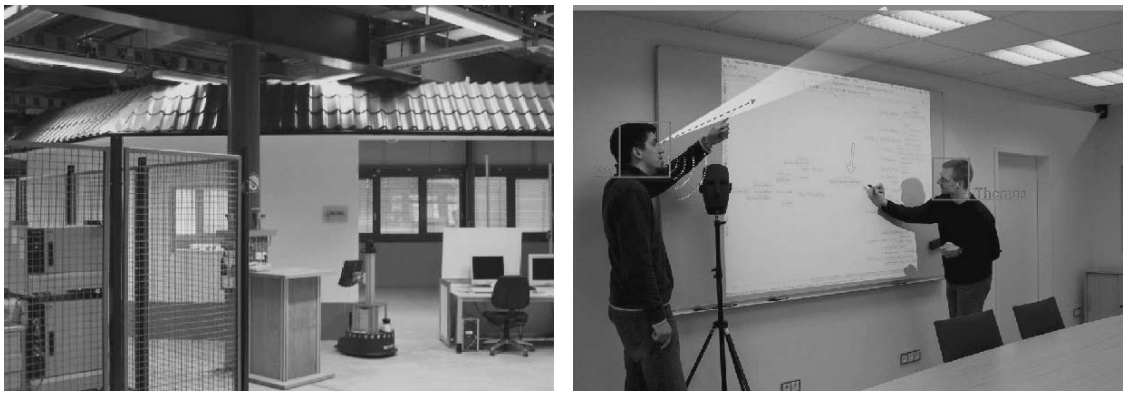

Fig. 1. Overview of the integration scenario—an Intelligent House, the FINCA, developed at the Robotics Research Institute of Technische Universität Dortmund (see text for description).

ing), the exclusive exploitation of such a color cue, unfortunately, results in a (very) large number of false positive predictions.

We developed an appearance-based approach to hand detection using scale-invariant salient region features (see [1] for an earlier version). Their evaluation appears promising since a pixel is considered as belonging to a hand only if it resides in a *structurally* "handlike" environment. The approach shows promising results, but still suffers from a rather large number of false positives. Thus, we furthermore focus on improving the robustness of this image-structure–based approach by integrating the aforementioned skin color information. The motivation for this is to take advantage of the fusion of two different image cues. Additionally, different fusion techniques are evaluated for the combination of both sources of information for reliable hand detection.

1.1. The FINCA

As an integration scenario of general pattern recognition techniques for intelligent multimodal human–machine interaction in dynamic environments—including gesture recognition based on the hand detection approach described in this article—we are developing an "Intelligent House"—the so-called FINCA (a Flexible Intelligent eNvironment with Computational Augmentation) [2]. The house is integrated into our laboratory at the Robotics Research Institute of Technische Universität Dortmund. Basically, the FINCA integrates two areas under one roof: a smart conference room and, connected to this, an open and flexible lab-space. Within both areas various sensors, namely cameras, microphones, infrared sensors, etc., are integrated. Electromechanical sensors (e.g., light switches) and actuators (e.g., light or sunblind control units) are integrated and connected via an EIB (European Installation Bus) installation. It is the most intuitive control of these devices we are aiming at, especially using gesture recognition techniques whose prerequisites are described in this article.

Ultimately an intelligent, cooperative house environment including service robots, which supports human users during various activities (conferences, information retrieval, communication, entertainment, etc.), is created. For natural and thus intuitive interaction with the house, special teaching of human users will not be required. Therefore, the FINCA detects, locates, and tracks communication partners by analyzing visual and acoustic data. The results are integrated allowing for multimodal scene analysis aiming at a successful automatic interpretation of the user's intentions.

In addition to its role as integration framework for various pattern recognition techniques developed at the Robotics Research Institute, the FINCA serves as a platform for scientific cooperations between researchers from different fields as well as between academia and industry.

Figure 1 gives an overview of the FINCA. On the upper left the house as a whole including our mobile robot is shown. The second image illustrates a typical (multimodal) interaction scenario within FINCA's smart conference room. All experimental evaluations described in this article have been performed within the FINCA environment.

1.2. Structure of the Article

The remainder of this article is organized as follows. In the subsequent section, the related work is briefly reviewed. Descriptions of current techniques for detecting hands in video images—*the* prerequisite for gesture recognition—are given including both structural and color cues as addressed by this article. In Section 3 the proposed hand detection approach integrating structural and color cues is discussed in detail. We give overviews of our techniques for hand detection with SIFT and skin color classification, respectively. Furthermore, the integration of both cues by sensor fusion is presented. Section 4 contains the description of the exper-

imental evaluation we performed demonstrating the effectiveness of the proposed approach.

2. RELATED WORK

Many different approaches to hand and limb detection using different kinds of visual cues have been proposed in recent years. A straightforward and simple approach that is often utilized (e.g., [3–5]) is to look for skin-colored regions in the image. Often, simple static color representations by histograms or mixture models (see, e.g., [6]) are used. Although this is practicable and efficient given controlled (or known) lighting conditions, skin color classification is difficult to handle under changing illumination. Color is directly influenced by the lighting conditions of the scene and is also dependent on the image acquisition hardware that is used. Thus, a color model that works fine for a given scenario may fail when the conditions change. Also, the choice of color space will normally have an effect on classification performance [7]. Instinctively, one would argue that a color space that has some invariance against intensity changes, i.e., that separates the color value from its brightness, would help to overcome some of the problems mentioned above. However, in [8] it is shown that this actually has only minor effects on the model quality and does not solve the general problem.

In order to deal with the difficulties of color classification in dynamic environments, several approaches to model adaptation have been proposed. This means the model parameters are adapted automatically to the current lighting conditions, as, e.g., in [9–11]. To achieve this, some kind of knowledge of the scene or the current lighting must be inferred from the images. Generally, the problem of model adaptation is not straightforward (for a recent related survey, see, e.g., [12]).

An obvious drawback of skin color classification is that other objects having skinlike colors in the chosen representation cannot be differentiated from real skin and therefore will yield false detections. So, given mostly unconstrained real-world scenarios, skin-color detection seems not reliable enough to be used as a stand-alone cue for hand detection.

Another widely used detection approach is to model an object by its shape, boundaries, or general appearance, i.e., based on image structure. Well-known examples are the appearance-based object detector of Viola and Jones [13] or Cootes' and Taylor's active appearance models [14]. However, for strongly articulated objects—like hands—showing a large variety of shapes, this is not feasible. Describing all possible appearances of a hand as a whole would either require a very flexible model (which very likely will be too general to be still reliable) or a huge model database that would be very difficult to handle. To overcome this difficulty when dealing with hand gestures, the amount of valid gestures is often limited to a rather small set of predefined poses (e.g., [4, 5]). Effectively, this means reducing the problem of unconstrained gestural interaction to recognition of a gestural command alphabet or sign language. In our research, however, we want the gestural interaction to be as unconstrained and natural as possible, which also implies that untrained users should be able to interact intuitively. Clearly, this is not the case if a command alphabet is defined, because a user has to be taught which gestures are meaningful.

A possible solution to this problem is given by "part models," as proposed, e.g., by Burl et al. [15]. This approach seems more promising for the task of structure-based hand detection since it models the object as a set of small characteristic parts (or image regions). The key assumption is that the local appearance of these parts will not change dramatically if the object is deformed or transformed. Thus, if the object is modeled as an assembly of regions (and some weak assumptions on spatial relationships are fulfilled), it may still be identified even under considerable deformations or changes in viewpoint. Another advantage is that partial occlusions of an object can be handled easily, since not all regions must be present to detect an object. As long as a sufficient number of parts can be found and several of them are spatially close, it can be concluded that the particular image area contains the object with high probability.

This leaves the question how to identify "characteristic parts" of the object and how to detect them in an image. A straightforward technique is to use standard appearance-based detectors (like the ones mentioned above) for the separate model parts and then judge the outcome (or find the best combination) by applying the shape constraints. The main drawback of this approach is that it needs to be analyzed in advance which parts of the object are characteristic and are big and salient enough to be detected reliably. It would be more feasible to find them automatically, i.e., to apply a salient local feature detection algorithm.

Several salient feature detectors and local region descriptors (i.e., local features) have been proposed. To find objects in arbitrary configurations, they should be invariant to changes in position, scale, and rotation of the object they describe. Furthermore, some invariance against affine distortions and illumination changes is required for realistic scenes. The best-known approach to this problem is probably the Harris corner detector [16]—which, however, is not scale invariant. More recent approaches include the salient regions detector of Kadir and Brady [17], the Speeded Up Robust Features (SURF) proposed by Bay et al. [18], and Lowe's Scale Invariant Feature Transform (SIFT) [19].

Given different types of cues, the task is to combine them into an overall classification scheme. A simple yet powerful approach is to concatenate the features into a higher dimensional feature vector and treat them as one. Another possibility is to determine the classification results for each feature separately and then use majority voting or fuse the results on a higher level. For

this multicue fusion, a great number of general approaches exist, the most straightforward of which are linear combiners (weighted sum, simple and weighted average) [20]. These are, however, only applicable if the results that should be combined are from the same domain, have the same dimensionality, and are normalized to the same dynamic range. Also, the weights have to be chosen in advance, which is often not straightforward.

Therefore, several approaches deal with learning the weights or with determining them automatically during run-time. For a comparison of linear combiners vs. common trained fusion rules, see [21]. A more complex dynamic combination scheme called Democratic Integration was proposed by Triesch and von der Malsburg [22]. Martin et al. [23] use Covariance Intersection to fuse detection results from different sensory inputs. Examples for the combination of different cues for hand detection are given in [24] (color and motion cues) and [25] (skin color and appearance-based body parts detector).

3. HAND DETECTION USING A COMBINATION OP SIFT AND COLOR FEATURES

In the following, we first describe the underlying parts of the classification approach, namely structural detection using SIFT, and skin color based filtering using Gaussian Mixture Models (GMM). Following this, the integration of both parts into a single classifier system is presented.

3.1. Hand Detection with SIFT

In Section 2 we stated that, while modeling hands as a whole using their appearance is bound to fail because of their strongly articulated nature, treating them as being composed of small characteristic parts is a promising approach. The question is now how to identify and detect such characteristic parts. We choose to use local descriptors that are computed at automatically determined salient feature points (often referred to as key-points). In the work presented here, we use the SIFT approach to extract structural features from the camera images. This is mainly because the method is well known and its potential has been shown in a number of different application fields (e.g., robot self-localization [26], camera calibration and scene reconstruction [27], and object-class recognition [28]). However, the local feature extraction routine for our approach may generally be chosen arbitrarily. We will give only a very brief overview of the SIFT algorithm (for details see [19]).

3.1.1. Feature extraction. SIFT is a staged approach, the first stage being the detection of salient key points. Key-point candidates are detected as local extrema of Difference-of-Gaussian (DoG) filters in a Gaussian Scale Space of the input image. These candidates are then subpixel interpolated. Key points showing low contrast or lying on edges are discarded, for they are not stable.

Invariance against scale and rotation is achieved by assigning a scale (according to the level of the scale space pyramid the key point was detected in) and an orientation (according to the principal orientation of gradients in a region around the key point) to each key point. Note that, in this step, points may be duplicated with different orientations if the local orientation histogram has multiple prominent peaks.

Finally, the local image descriptor (i.e., the features) is calculated as a collection of smoothed histograms of gradient orientations and magnitudes over the local image region. The size of the feature vector depends on the number of histograms and the number of bins in each histogram. In Lowe's original implementation, a 4-by-4 patch of histograms with 8 bins each is used, yielding a 128-dimensional feature vector. We use a MATLAB/C implementation of the SIFT algorithm provided by Vedaldi [29].

3.1.2. Matching. Given an image of the scene, we obtain a (typically large) number of SIFT features describing salient points in the image. In order to detect hands, we build a database containing descriptors extracted from many images containing hands in different configurations (see Section 4.1 for details on the data). We also build a large database of background descriptors from images taken inside the FINCA. Following Lowe's proposal for object recognition, we implement the matching algorithm as follows: Let k_i be the key point descriptor that should be classified. Let d_{fg} and d_{bg} be the Euclidean distances to the nearest neighbors (found by means of a kd-tree [30] search) to k_i from the foreground and background databases, respectively. We decide on the key point being fore- or background by thresholding the classification score s_{class}, which is the ratio of the two distances d_{fg} and d_{bg}.

3.1.3. Candidate filtering. With the above matching process, we obtain a label for each key point in the input image. To achieve a high number of true positives, we have to choose a classification threshold t_{class} on s_{class} that is larger than 1; i.e., we actually allow for key points to be classified as positives even if a slightly better match has been found in the background database. Obviously, this results in a large number of false positives, a drawback that has also been reported by Lowe for matching on key-point level. This is why Lowe suggests matching groups of features using a generalized Hough transform followed by a detailed geometric fit [19]. While this yields excellent results for rigid object detection, it is not applicable to the problem of hand detection, again due to the strongly articulated nature of hands and because we expect to see them in arbitrary configurations and viewing angles. We have to apply a weaker constraint on spatial configuration.

Because SIFT yields a large number of key points (around 1000 for a typical image from our sample set),

for all keypoints $k_1 \ldots k_n$ labeled as positives **do**
list = getKeypointList(k_i)
count positives n_{pos} and negatives n_{neg} in *list*
if $\{n_{pos} \geq n_{pos,\,min})\&(\frac{n_{pos}}{n_{neg}} \geq f_{min})$ **then**
accept k_i as true positive.
else if $(n_{pos} < n_{pos,\,min})\&(\frac{n_{pos}}{n_{neg}} < f_{min})$ **then**
reject k_i
else
find the m keypoints $l_1 \ldots l_m$ closest to k_i
if $l_1 \ldots l_m$ are all true positives **then**
accept k_i as true positive
else
reject k_i
end if
end if
end for

function getKeypointList(*candidate*)

(a) *list* = all points in circular region around *candidate* with $r = \sigma \cdot c$, $c = const$, σ = candidate scale
(b) *list* = all point in circular region around *candidate* with r = c, $c = const$
(c) *list* = n spatially closest points to *candidate*
return *list*

Fig. 2. Outline of the candidate filtering algorithm.

we assume that typically we will find multiple key points on hands of which most will be classified as foreground (i.e., we have a low false negative rate), whereas most false positives will be scattered over the image and thus will be surrounded by numerous true negatives. In other words, we expect the true positives to form spatial clusters while false positives will often be isolated, and consequently use this as constraint to eliminate false positives. This leads to an efficient candidate filtering algorithm by analyzing lists of key points spatially connected to the candidates in question. The outline of the algorithm is given in Fig. 2.

The first step is to determine the adjacency list for each (positive) keypoint candidate. Three different approaches were implemented and evaluated: Circular regions of fixed size centered around the candidate; circular regions with sizes proportional to the SIFT scale of the candidate; and taking the n spatially nearest neighbors. We then evaluate points based on the total number of positive candidates in their respective list (threshold $n_{\text{pos, min}}$) and the ratio between positives and negatives (threshold f_{min}). If a candidate passes both criteria, it is accepted as a foreground point. If it fails on both criteria, it is rejected. Otherwise, it is further evaluated in a second step by determining the number of positives m it is connected to.

This approach discards a large number of false positives and yields promising results. Still, however, it cannot handle clusters of false positives and fails to reject them. Figure 3 shows two example results.

3.2. Skin Color Classification

Since most approaches relying on a single type of feature exhibit drawbacks in specific fields or under certain circumstances, it is intuitive to think about the integration of different features. Provided that the different cues show substantially different characteristics and behavior, the assumption is that the strengths of one of them can compensate for the weaknesses of others, and vice versa. Consequently, this should yield a system that shows better results, reliability, and robustness than any of the respective subsystems. In our scenario, we have a structure-based region classifier for hand detection that typically suffers from a large number of false positives (see Section 3.1). Despite the drawbacks of skin color detection (as discussed in Section 2), it

Fig. 3. Examples for SIFT based hand detection results. Sift key points are depicted by squares, white squares represent positives, black squares are negatives. Left: very good detection, the hands are identified correctly and almost all false positives are discarded. Right: here, the filtering algorithm fails because the false positives form large clusters.

Some example results using the complete filtering algorithm. $n_{\text{pos, min}}$ abbreviated to n. RS10: region size 10σ. R15: fixed region size 15. NN16: 16 nearest neighbors. The values in brackets give the relative drop for the true and false positive rates compared to NN matching with the same threshold t_{class}

	Region	t_{class}	n	f_{min}	m	% TP (Δ*NN*)	% FP (Δ*NN*)
1	RS10	2.0	3	1.25	1	85.08 (–10.2)	4.04 (–52.3)
2	RS10	2.5	3	1.00	2	91.55 (–6.2)	6.32 (–40.8)
3	R15	2.0	4	1.25	1	88.18 (–6.9)	4.05 (–52.2)
4	R15	2.5	3	1.00	2	93.07 (–4.6)	6.00 (–43.8)
5	NN16	2.0	4	1.25	1	90.26 (–4.7)	4.34 (–48.8)
6	NN16	2.5	3	1.50	2	91.30 (–6.4)	5.61 (–47.4)

seems reasonable to utilize a skin-color classifier because it will reliably reject those false positives that lie on non–skin-colored regions. Plus, a nonadaptive color model may be implemented as a simple look-up table once it has been trained, and thus adds almost no additional computational costs. However, the answer to the question how the different features should be combined to achieve the best result is, in general, not straightforward. In Section 3.3, we will investigate different alternatives.

For the purpose of skin color detection, we utilize a simple approach using Gaussian Mixture Models (GMMs): The training samples are first clustered using the k-means algorithm and then the clusters are approximated by Gaussian distributions. The final model thus consists of several multivariate Gaussians representing the sample distribution. Images are classified pixelwise by calculating the model scores for the pixel color values and then determining the class of the "best" mixture. Obviously, as mentioned above, this can be done by generating a look-up table using a color dummy file.

The models were trained on a small set of training images (see Section 4.1). Two mixtures were trained separately for fore- and background, respectively, and then combined into a single model. We investigated different color spaces and mixture sizes in order to find the best combination for our data.

Note that, in our scenario, the conditions for skin color detection can be very challenging since some cameras directly face the windows (glare effects, with most of the objects in front of the window appearing almost black) and the wood inlay of the furniture has a skin-like color (see Fig. 5). This results—consistently for all tested color spaces—in a rather large error rate for skin pixels, since the background set contains many skinlike samples.

3.3. Data Fusion

Given the above-mentioned structural and color cues, the task is to combine them into a single classifier system which can be used for robust hand detection in video images. In this article we investigated three different fusion approaches.

A straightforward approach is to incorporate the skin color map computed using GMMs within a preprocessing step. Here, a binary skin map is computed and used to mask the input image. Subimages enclosing nonzero image regions are extracted, and SIFT-based hand detection is applied to these subimages only (instead of the whole image). Compared to most other approaches, this has the additional advantage of considerably accelerating the matching process, since large portions of the input image do not have to be treated. Alternatively (but principally identically), fusion can be performed as postprocessing where the SIFT key points are weighted by their skin color probability.

The second integration approach we considered combines both features in a single (higher dimensional) feature vector prior to classification. This is done as follows: First, we calculate a color histogram over an image region specified by the key-point scale (meaning we actually do not use the GMM skin classifiers for this approach). This histogram is then vectorized, normalized to unit length, and attached to the SIFT descriptor. Classification is performed as described in Section 3.1.2 using the compound feature vector. Of course, the database entries used for NN matching must then be constructed in the same way.

Our third approach incorporates the combination of saliency maps, which are calculated separately for both information cues. Every positive key point (either SIFT or skin color related) serves as origin of a single Gaussian. For the SIFT saliency map, the variance of the Gaussian is dependent on the particular scale of the SIFT key point. For the skin map, a fixed variance is used. Figure 4 shows an example. The two saliency maps are then fused. The resulting combined map is thresholded, and all hand candidates having saliency values below the threshold are rejected.

4. EXPERIMENTAL EVALUATION

In order to evaluate the effectiveness of the new approach for the detection of hands in video images proposed in this article, we conducted various practical experiments. Therefore, human users of our intelligent house (see Section 1.1) were asked to perform certain

Fig. 4. Examples for saliency maps. From left to right, top to bottom: original image, SIFT saliency, skin saliency (L*a*b* classifier), skin saliency (nRG classifier).

Fig. 5. Typical examples of camera images used.

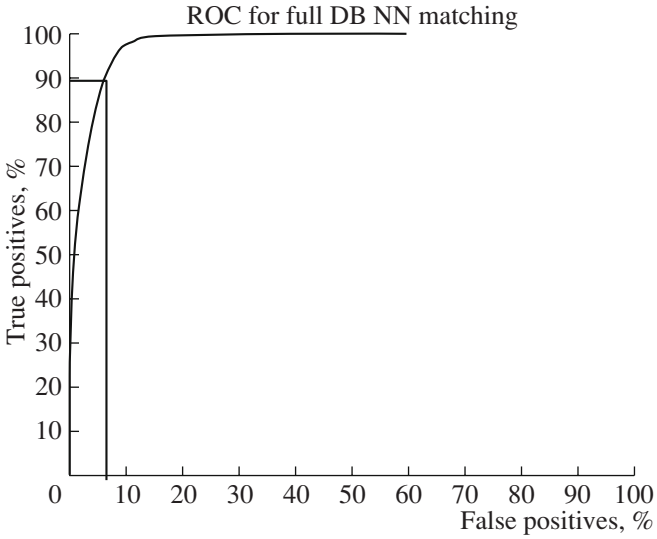

Fig. 6. Effectiveness of the exclusive application of the structural cue for hand detection in still video images: The ROC curve illustrates the results for nearest neighbor matching using the complete training database of SIFT descriptors.

gestures in ordinary situations within the smart conference room of the FINCA. By means of the ceiling-mounted video cameras, still images capturing the gestures were taken and, subsequently, analyzed using the techniques presented in the previous sections.

In the following the results of the experiments are presented in detail. First the datasets used are described. After that results for the SIFT-based detection approach as well as for the combined approach integrating both structural and color cues are given.

4.1. Datasets

For training and testing, we recorded a dataset of 466 color images with PAL resolution. The set was recorded inside our intelligent house using two cameras covering different views of the room over different days and under varying lighting conditions. It contains images of 4 different people wandering around inside our smart conference room and gesticulating. Note that we did not constrain the type of poses or gestures performed, that the people appeared at different distances to the cameras, and that they were allowed to move around freely in the camera's field of view. Figure 5 shows some example images.

These images were segmented into hand and non-hand parts manually, where a small region around the perimeter of hands is also labeled as belonging to the hand to account for SIFT descriptors that describe typical hand regions, but lie outside the actual skin area. From this set, 145 images were randomly selected for testing. The remaining 321 images were taken for training of our classifier. GMM-training for skin color classification (Section 3.2) was performed on an alternative set of 46 sample images from the same scenario.

The final database of SIFT descriptors applied for the evaluation of the structural cue contains approximately 200 000 entries for the background, and 8700 for the foreground (i.e., the hands).

4.2. SIFT-based Hand Detection

In the first part of the experimental evaluation, we concentrated on the effectiveness of the SIFT-based approach for hand detection in still video images; i.e., the structural cue was used exclusively (see also [1]). We, therefore, applied variants of our detection approach as described in Section 3 aiming at the evaluation of the approach in general and of the effectiveness of the proposed rejection criterion, respectively.

Using the database of SIFT descriptors extracted from training images (see previous section), first, the standard nearest neighbor (NN) matching technique (implemented as kd-tree search for speedup) as described in [19] is applied to the set of test images. The goal of this specific evaluation is to investigate whether SIFT descriptors are suitable in general for the detection of articulated objects like hands in unstructured images originating from real world scenarios. Note that SIFT was originally developed for the detection of rigid, i.e., nonarticulated objects in images where those objects known from training samples might occur as scaled or rotated instances. As mentioned before, in our scenario no constraints with regard to the appearance of the hands to be detected in the images are given.

The ROC curve for NN matching using the full database of training examples is shown in Fig. 6, the variational parameter being the threshold t_{class} on the distance ratio s_{class}. It can be seen that, generally, the original SIFT approach for rigid object detection using simple NN matching already yields satisfactory results. To get a high number of true positives, we allow for the accepted points to have a distance ratio $s_{\text{class}} > 1.0$, which means they are in fact more similar to some of the background examples. The point marked with 90% true positives[2] and 6.8% false positives corresponds to a threshold of 1.7. Due to the large number of key points that are identified by SIFT (typically 800–1600 per image), this results in a large number of false positives which typically lie on foreground objects (e.g., the person's body) not represented in the database. However, a considerable number of these will be discarded by our filtering algorithm.

By means of the proposed rejection scheme, i.e., the modification of the original SIFT-based detection, the above-mentioned reduction of false positive classifications is addressed. As described in Section 3, this rejection scheme consists of a hyteresislike approach where

[2] We assume that, for our application, a true positive rate lower than 85 to 90% will not be sufficient.

a list of key points spatially connected to the candidate in question is determined. Based on a two-stage filtering technique, the number of false positives is considerably reduced.

In a set of experiments, we evaluated the influence of the three parameters $n_{\text{pos, min}}$, f_{min}, and m using different classification thresholds t_{class} for the NN matching stage and several region sizes. Basically, the algorithm (see Fig. 2) contains three different (technical) definitions of spatial neighborhood resulting in regions:

(1) defined depending on the particular SIFT scale,

(2) of fixed sized, or

(3) containing the k nearest neighbors.

By analyzing the particular ROC curves, the experimental evaluation turned out that all three variants reduce the number of false positives successfully. Basically, scale-based and fixed regions do not differ substantially with regard to the reduction rates, which, on the one hand, seems surprising. Reconsidering the setting of the evaluation, it becomes clear that this behavior seems to be an artifact of the sample set analyzed. The room the images were recorded in is quite small, and so the assumption that hand sizes do not vary strongly holds for most cases. Given a different scenario, a negative effect is very likely when using a fixed region size. Detailed results for the above-mentioned variants of determining the particular key-point lists are given in [1].

The third variant for candidate filtering does not use explicit regions; instead we generate the input list using the k spatially closest key points to the candidate. Figure 7 shows the ROC curves for this approach. Compared to the aforementioned definitions of regions surrounding a particular key point, our filtering technique analyzing the k nearest neighbors performed best since we do not make assumptions on appropriate region sizes, but evaluate the same number of neighboring points for each candidate.

Reconsidering the overall algorithm (see Fig. 2), it can be seen that the criteria evaluated so far—the minimum required number of positives $n_{\text{pos, min}}$ and the minimum ratio of positives and negatives f_{min}—both belong to the first stage and are applied simultaneously. However, failing one of them is not sufficient to reject a candidate key point. Instead, we reject only key points that fail on *both* conditions, and we only accept those that satisfy both. The remaining, which pass one criterion, but fail on the other, are further evaluated in a second stage.

In this second stage of the filtering approach, the m spatially closest neighbors of the candidate point are considered. A key point is only accepted as true positive if *all* m neighbors are true positives, which means they must all have passed both conditions in the first stage. Setting m to a high value will eliminate "isolated" positives, but will also tend to discard key points at the margins of positive clusters. Figure 8 shows the ROC

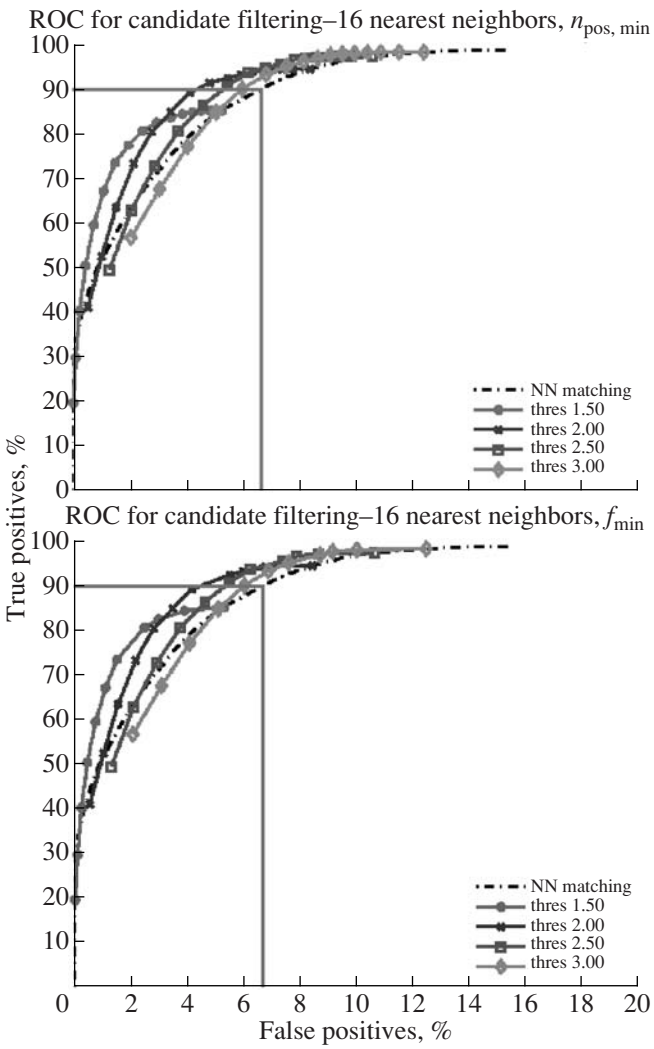

Fig. 7. ROC curves for candidate filtering in hand detection applying SIFT only using $k = 16$ nearest neighbors. Top: variating $n_{\text{pos, min}}$. Bottom: variating f_{min}.

curves for different parameter combinations, variating m in a reasonable range (from 0 to 10). The plot has been scaled for better recognizability. Note that these curves have two fixed ends that are defined by the outcome of the first filtering stage: The starting point corresponds to the complete set of candidates that passed one of the two initial conditions, the end point to the number of candidates that passed both conditions. It can be seen that this is a pretty strong criterion, since for all values of $m > 0$, a certain portion of true positives is rejected. However, the effect on false positives is stronger. Since most false positives are already rejected for $m = 1$, and higher values for m will only discard more true positives, we will only take into consideration values of 1 and 2 for m for the evaluation of our complete system.

The table shows the results of a few example runs using parameter sets that seem reasonable based on the

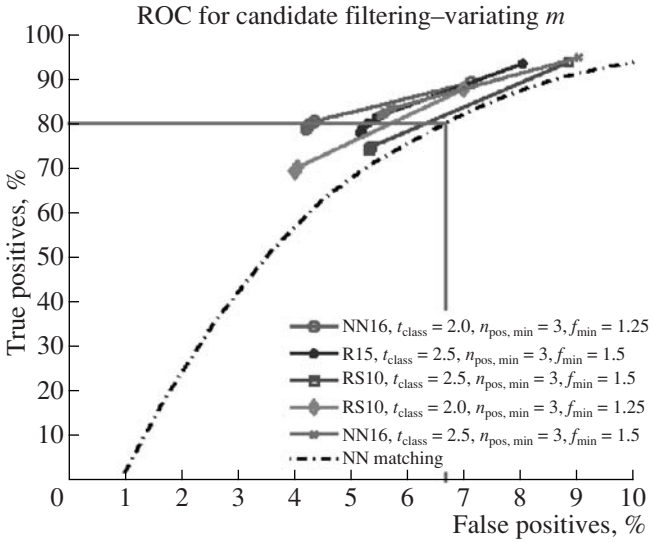

Fig. 8. ROC curves for candidate filtering in hand detection applying SIFT only variating *m* with different parameter sets. NN16: 16 nearest neighbors. R15: fixed region size 15. RS10: region size 10σ.

aforementioned evaluation results of the different stages. For almost all parameter combinations, the filtering approach achieves a substantial reduction of the number of false positives while retaining true positive rates only slightly lower than in the initial NN matching stage. The best combinations reduce the number of false positives by one-half, while only dropping around 5% of the true positives. This is an acceptable tradeoff, since the required rate of approximately 90% true positives can still be achieved in most cases.

4.3. Detection of Hands Using Cue Integration

In the previous sections it was shown that by means of SIFT-descriptors reliable hand detection in video images of realistic HMI scenarios is possible. We, furthermore, demonstrated that the rather high number of false positive detections as produced when applying the original SIFT approach can substantially be reduced using the proposed hysteresislike filtering technique.

The basic idea of the hand detection method described in this article is, however, not limited to the exploitation of the structural cue. Instead, color information is integrated aiming at a further reduction of the number of false predictions, which is especially necessary for robust gesture recognition as an intuitive input modality in human–machine interaction applications. Note that the exclusive use of approaches based on skin-color classification is, due to the complex setting in our intelligent house environment, not suitable for our scenario.

We trained our skin models on 46 image samples in full PAL-resolution (yielding a total of more than 19.3 million pixels) recorded within our target scenario. The data was manually labeled with respect to skin color. In summary, the dataset contained 17.7 million background pixels and 1.6 million skin pixels. These were used to separately train mixture models for skin and background, which were then combined into a single model. Eight different color spaces were investigated, and the number of Gaussians in the models was varied between 5 and 150. We will skip the detailed evaluation results for the whole process (because this would go beyond the scope of this article) and only report the best results that were achieved.

We found the classifiers trained in the L*a*b and normalized RG (nRG) color space to work best. For robust detection of skin colored regions, a small number of mixtures (L*a*b: 5 for skin, 16 for background; nRG: 5/2) was found to be sufficient. On a test data set containing 13 million pixels, the L*a*b* classifier achieved an overall classification error of 2.8%. While only 0.7% of background pixels were classified incorrectly, the error rate for skin pixels was 53%. For the nRG classifier, the overall error rate was 10%, which, in comparison, is rather high. However, this classifier showed the best results for skin pixels (32% error rate). So, we have two classifiers showing substantially different behavior on the test set: The L*a*b* classifier is "pessimistic," striving for a low overall error rate and accepting a large number of false negatives to achieve this, while the nRG classifier is more "optimistic" and trades in a good skin recognition rate for a higher number of false positives.

As discussed in Section 3.3, different schemes for fusing the data from both the SIFT-based structural cue and the skin-color cue are investigated.

First experiments showed that using binary (morphologically closed) skin maps as a pre- or postprocessing step is not suitable for robust hand detection. This is because many of the structural descriptors that describe hands lie in fact outside the skin area. Also, since a skin color segmentation will never be perfect and will still exhibit holes inside skin areas, some true positives lying on skin will also be discarded. Thus we concentrated on the latter two integration methods.

In Fig. 9 the results of the evaluation of enhanced SIFT descriptors are presented. We evaluated certain variants for descriptor enhancements using different color space models and granularities of the appropriate histograms, namely HSV (h32 and hs8 × 8), normalized RG (n-rg8 × 8), and LAB ((l)a*b*8 × 8). The dimensionality of the original descriptor vectors (128) was enhanced by the particular histogram sizes. It can be seen that, although the integration technique works in principle, only slight improvements over the SIFT-only detection (denoted as kd-tree where the name originates from the actual implementation of the matching technique as a kd-tree) can be achieved. In fact, in some cases the performance was even worse than the reference. The reason for this lies in the missing flexibility

of color histograms with regard to illumination changes (which are included in the—realistic—sample-set used).

The third type of fusing both sources of information is based on the integration of saliency maps each calculated separately for the SIFT- and the color-cue, respectively. In order to actually combine the probability maps, basically, different strategies can be used. In our experiments we focused on the pixelwise (weighted) summation of saliency values.[3] We also tried multiplying the maps which, however, did not improve the results and when using the L*a*b color space produced even worse results. This is mainly understood by the fact that in this case multiplying saliency maps suffered from the rather pessimistic behavior of the particular skin classifiers skipping too many true positives.

In Fig. 10, ROC curves for the summation of SIFT-based and n-RG based (top), or L*a*b based (bottom) skin color saliencies are shown. Again the reference curve representing the results obtained when evaluating SIFT-descriptors only is denoted as kd-tree (see explanation in previous section). For the saliency combination, three different curves are given where the SIFT thresholds for NN classification are modified. By analyzing the ROC curves, it can be seen that the integration of structural and color cues by combining related saliency maps greatly reduces the number of false positive detections while still very high true positive rates can be achieved.

To summarize in Fig. 11 we show the best results for both skin classifiers based on nRG and L*a*b color space, respectively. The variational parameter is the classification threshold on the combined saliency map as described earlier. We marked two "working points" with a true positive rate of 90 and 94%. In the first case the false positive rate could be reduced from 6.3 to 4.2% (a relative reduction by 33%) using the L*a*b colorspace, and to 4.5% (relative 28%) using nRG. In the latter case the reduction is from 7.5 to 5.0% (33%) and to 5.4% (28%), respectively.

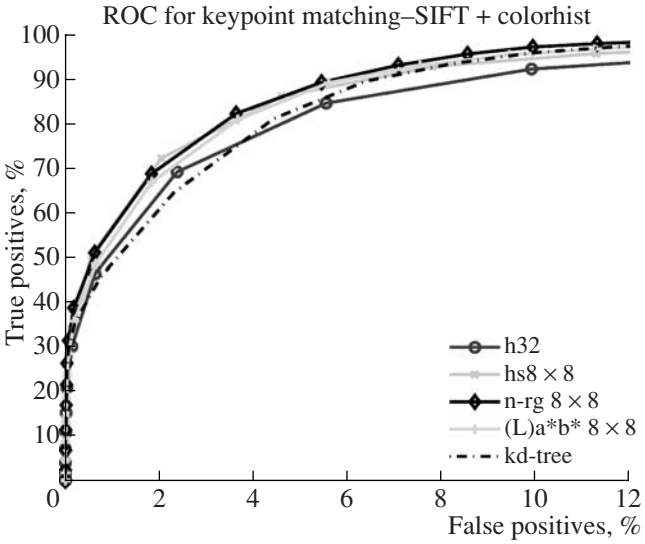

Fig. 9. ROC curves for hand detection experiments using enhanced SIFT descriptors additionally integrating color histograms of the particular key-points regions, kd-tree: reference using original SIFT descriptors; h32: additional incorporation of hue-based color histograms (from HSV color space) with 32 bins; hs8 × 8: integrated color histogram is based on hue and saturation (from HSV color space) and calculated on 8 × 8 bins; n-rg8 × 8: same as hs8 × 8 but using normalized RG color space; (L)a*b*8 × 8: dito for LAB space.

5. CONCLUSIONS

The basic motivation for the development of so-called "intelligent systems" for human machine interaction (HMI) applications is to allow for most intuitive and, thus, easy usability of technical systems. Humans usually consider a system smart if it shows reasonable reactions to their actions related to the services offered. An important aspect for the acceptance and usefulness of such intelligent systems is the naturalness of interfaces it offers.

The key modalities used by humans for "natural" interaction with technical systems are speech and gestures. In order to allow for gesture recognition including both dynamic and static (e.g., pointing) gestures the robust detection of hands in video images is a major prerequisite. In this article we presented an approach for robust hand detection in still video images covering realistic scenarios.

In our work we, generally, focus on the recognition of unconstrained gestures performed in (almost) arbitrary environments. Due to their rather limited performance for related still video images, especially originating from scenarios with different lighting conditions, approaches based on standard skin-color classification for hand detection cannot be used exclusively. Thus, we developed a detection approach which is based on the exploitation of two different sources of information, namely a structural cue and a skin-color cue.

For the structural cue integrated into our overall detection algorithm, SIFT-descriptors are used. This initial classification step is based on a database of descriptors that are trained on sample images containing either hands or background. By means of a hysteresislike filtering technique, the number of false positive classifications can be limited reasonably. In order to further reduce the number of misclassifications, the approach proposed in this article additionally contains a skin-color classification stage based on Gaussian Mixture Models. By means of saliency maps derived from both cues, data fusion is performed.

We demonstrated the effectiveness of our approach in a detailed experimental evaluation on a challenging

[3] In the experiments described in this paper the particular saliencies were equally weighted.

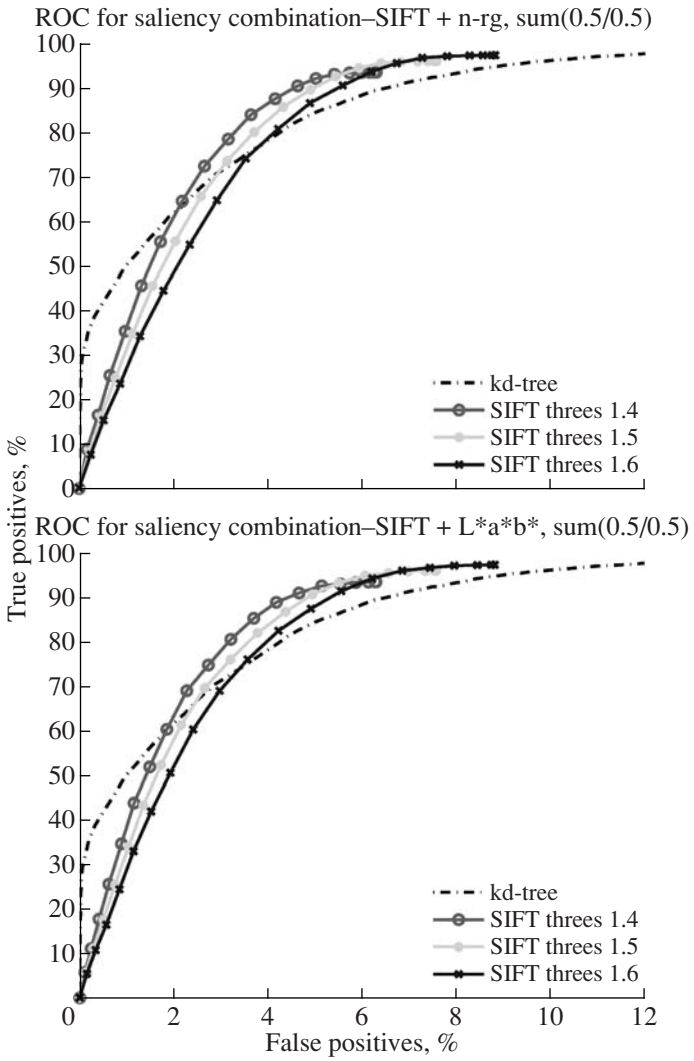

Fig. 10. ROC curves illustrating the efficiency of the combined hand detection approach using SIFT and skin color cues integrated by summation of derived saliencies (top: n-RG color space; bottom: L*a*b color space used).

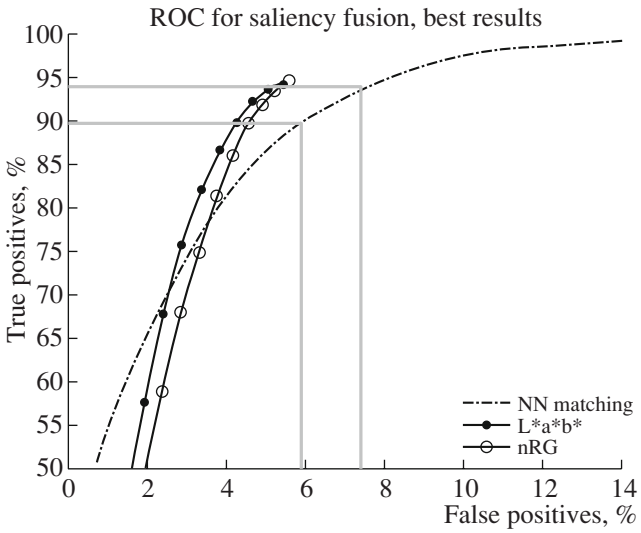

Fig. 11. Best results for saliency fusion using L*a*b and nRG(B) color space for classification.

task related to human–machine interaction in our intelligent house, the FINCA. For different lighting conditions, hands could be detected robustly in images covering different views of our smart conference room, where different people were wandering around and gesticulating in an unconstrained manner.

The major outcome of the developments presented in this article, and thus, the main contribution of our work, is the realization of an important initial stage for actual gesture recognition. Consequently, intelligent human–machine interaction applications exploiting this intuitive modality in related domains can benefit from the approach presented.

REFERENCES

1. J. Richarz, T. Plötz, and G. A. Fink, "Detecting Hands in Video Images Using Scale Invariant Local Descriptors," in *Proceedings of IASTED Int. Conf. on Visualization,*

Imaging and Image Processing (VHP 2007) (Palma de Mallorca, Spain, 2007), pp. 259–264.

2. T. Plötz, "The FINCA: A Flexible, Intelligent eNvironment with Computational Augmentation," http://www.finca.irf.de, 2007.
3. N. Hofemann, J. Fritsch, and G. Sagerer, "Recognition of Deictic Gestures with Context," in *Proceedings of 26th Deutsche Arbeitsgemeinschaft Mustererkennung Symposium* (LNCS Vol. 3175, Springer, 2004), pp. 334–341.
4. R. Lockton and A. W. Fitzgibbon, "Real-Time Gesture Recognition Using Deterministic Boosting," in *Proceedings of British Machine Vision Conference, 2002*, pp. 817–826.
5. J. Triesch and C. von der Malsburg, "Classification of Hand Postures against Complex Backgrounds Using Elastic Graph Matching," Image and Vision Computing **20**, 937–943 (2002).
6. M. J. Jones and J. M. Rehg, "Statistical Color Models with Aapplication to Skin Detection," International Journal of Computer Vision **46** (1), 81–96 (2002).
7. S. Jayaram, S. Schmugge, M. C. Shin, and L. V. Tsap, "Effect of Colorspace Transformation, the Illuminance Component, and Color Modeling on Skin Detection," in *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, 2004*, Vol. 2, pp. 813–818.
8. M. C. Shin, K. I. Chang, and L. V. Tsap, "Does Colorspace Transformation Make Any Difference on Skin Detection?," in *Proceedings of 6th IEEE Workshop on Applications of Computer Vision, 2002,* pp. 275–279.
9. F. Dadgostar and A. Sarrafzadeh, "An Adaptive Real-Time Skin Detector Based on Hue Thresholding: A Comparison on Two Motion Tracking Methods," Pattern Recognition Letters **27**, 1342–1352 (2006).
10. Q. Zhu, K.-T. Cheng, and C.-T. Wu, "A Unified Aadaptive Approach to Accurate Skin Detection," in *Proceedings of IEEE Int. Conf. on Image Processing, 2004*, Vol. 2, pp. 1189–1192.
11. L. Sigal, S. Sclaroff, and V. Athitsos, "Skin Color-Based Video Segmentation under Time-Varying Illumination," IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (7), 862–877 (2004).
12. P. Kakumanu, S. Makrogiannis, and N. Bourbakis, "A Survey of Skin-Color Modeling and Detection Methods," Pattern Recognition **40** (3), 1106–1122 (2007).
13. P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," in *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, 2001*, pp. 511–518.
14. T. F. Cootes and C. J. Taylor, "Statistical Models of Appearance for Medical Image Analysis and Computer Vision," *Proceedings of SPIE Medical Imaging, 2001*, Vol. 4322, pp. 238–248.
15. M. C. Burl, M. Weber, and P. Perona, "A Probabilistic Approach to Object Recognition Using Local Photometry and Global Geometry," *Proceedings of 5th European Conf. on Computer Vision, LNCS 1407, 1998,* Vol. 2, pp. 628–641.
16. C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Proceedings of Alvey Vision Conference, 1988*, pp. 147–151.
17. T. Kadir and M. Brady, "Scale, Saliency and Image Description," Int. Journal of Computer Vision **45** (2), 83–105 (2001).
18. H. Bay, T. Tuytelaars, and L. van Gool, "Surf: Speeded up Robust Features," in *Proceedings of 9th European Conf. on Computer Vision* (LNCS Vol. 3951, Springer, 2006), pp. 404–417.
19. D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," Int. Journal of Computer Vision **60**, 91–110 (2004).
20. G. Fumera and F. Roli, "Linear Combiners for Classifier Fusion: Some Theoretical and Experimental Results," in *Proceedings of Int. Workshop on Multiple Classifier Systems, LNCS 2709, 2003*, pp. 74–83.
21. F. Roli, J. Kittler, G. Fumera, and D. Muntoni, "An Experimental Comparison of Classifier Fusion Rules for Multimodel Personal Identity Verification Systems," in *Proceedings of Int. Workshop on Multiple Classifier Systems, LNCS 2364, 2002*, pp. 325–336.
22. J. Triesch and C. von der Malsburg, "Democratic Integration: Self-Organized Integration of Adaptive Cues," Neural Computation **13**, 2049–2074 (2002).
23. C. Martin, E. Schaffernicht, A. Scheidig, and H.-M. Gross, "Sensor Fusion Using a Probabilistic Aggregation Scheme for People Detection and Tracking," in *Proceedings of 2nd European Conf. on Mobile Robots, 2005*, pp. 176–181.
24. J. Alon, V. Athitsos, and S. Sclaroff, "Simultaneous Localization and Recognition of Dynamic Hand Gestures," in *Proceedings of IEEE Workshop on Motion and Video Computing, 2005,* Vol. 2, pp. 254–260.
25. A. S. Micilotta, E.-J. Ong, and R. Bowden, "Real-Time Upper Body Detection and 3D Pose Estimation in Monoscopic Images," in *Proceedings of European Conference on Computer Vision* (LNCS 3953, Springer Verlag, 2006), pp. 139–150.
26. J. Kosecka and F. Li, "Vision Based Topological Markov Localization," in *Proceedings of IEEE Int. Conf. on Robotics and Automation, 2004,* Vol. 2, pp. 1481–1486.
27. J. Liu and R. Hubbold, "Automatic Camera Calibration and Scene Reconstruction with Scale-Invariant Features," in *Proceedings of Int. Symposium on Visual Computing, LNCS 4291, 2006*, pp. 558–568.
28. D. A. Lisin, M. A. Mattar, and M. B. Blaschko, "Combining Local and Global Image Features for Object Class Recognition," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2005,* pp. 47–54.
29. A. Vedaldi, "Matlab Sift Implementation," http://vision.ucla.edu/vedaldi/code/sift/sift.html.
30. J. L. Bentley, "Multidimensional Binary Search Trees Used for Associative Searching," Communications of the ACM **18** (9), 509–517 (1975).

Thomas Plötz received a diploma in technical computer science from the University of Cooperative Education, Mosbach, Germany, in 1998. He received a diploma and the PhD degree (Dr.-Ing.) in computer science from the University of Bielefeld, Germany, in 2001 and 2005, respectively.

From 2001 to 2006 he was with the Applied Computer Science Group at the Faculty of Technology of Bielefeld University. In 2006 he joined the Intelligent Systems group at the Robotics Research Institute of the Dortmund University of Technology, Germany, where he holds a post-doctoral research position.

Dr. Plötz is interested in general aspects of machine learning and pattern recognition techniques and applications for various domains like speech-processing, automatic recognition of handwritten script, image processing, or bioinformatics. He is coordinating various research activities within the Smart environment project at the Robotics Research Institute's "Intelligent House"—the FINCA.

Jan Richarz received a diploma in computer engineering from the Ilmenau Technical University, Germany, in 2006.

He joined the Intelligent Systems group at the Robotics Research Institute of the Dortmund University of Technology, Germany, in May 2006. There, he is working within the Institute's smart environment project FINCA.

His research interests include pattern recognition techniques for image processing, object identification and feature extraction, and multimodal human–machine interaction.

Genot A. Fink received a diploma in computer science from the University of Erlangen-Nürnberg, Erlangen, Germany, in 1991 and the PhD degree (Dr.-Ing.) also in computer science from Bielefeld University, Germany, in 1995. In 2002 he received the venia legendi (Habilitation) in applied computer science from the Faculty of Technology of Bielefeld University.

From 1991 to 2005 he was with the Applied Computer Science Group at the Faculty of Technology of Bielefeld University. Since 2005 he is professor for Pattern Recognition in Embedded Systems at the Dortmund University of Technology, Germany, where he also heads the Intelligent Systems group at the Robotics Research Institute (IRF).

His research interests lie in the development and application of pattern recognition methods in the fields of man–machine interaction, multimodal machine perception including speech and image processing, statistical pattern recognition, handwriting recognition, and the analysis of genomic data.

Dr. Fink is Senior Member of the Institute of Electrical and Electronics Engineers (IEEE), the IEEE Signal Processing Society, and the IEEE Computer Society.