Vision-based Hand Gesture Recognition for Human-Computer Interaction

X. Zabulis†, H. Baltzakis†, A. Argyros‡†

†Institute of Computer Science
Foundation for Research and Technology - Hellas (FORTH)
Heraklion, Crete, Greece

‡Computer Science Department, University of Crete
Heraklion, Crete, Grece

{zabulis,xmpalt,argyros}@ics.forth.gr

1 Introduction

In recent years, research efforts seeking to provide more natural, human-centered means of interacting with computers have gained growing interest. A particularly important direction is that of *perceptive user interfaces*, where the computer is endowed with perceptive capabilities that allow it to acquire both implicit and explicit information about the user and the environment. Vision has the potential of carrying a wealth of information in a non-intrusive manner and at a low cost, therefore it constitutes a very attractive sensing modality for developing perceptive user interfaces. Proposed approaches for vision-driven interactive user interfaces resort to technologies such as head tracking, face and facial expression recognition, eye tracking and gesture recognition.

In this paper, we focus our attention to vision-based recognition of hand gestures. The first part of the paper provides an overview of the current state of the art regarding the recognition of hand gestures as these are observed and recorded by typical video cameras. In order to make the review of the related literature tractable, this paper does not discuss:

- techniques that are based on cameras operating beyond the visible spectrum (e.g. thermal cameras, etc),
- *active techniques* that require the projection of some form of structured light, and,

- *invasive techniques* that require modifications of the environment, e.g. that the user wears gloves of particular color distribution or with particular markers.

Despite these restrictions, a complete review of the computer vision-based technology for hand gesture recognition remains a very challenging task. Nevertheless, and despite the fact that the provided review might not be complete, an effort was made to report research results pertaining to the full cycle of visual processing towards gesture recognition, covering issues from low level image analysis and feature extraction to higher level interpretation techniques.

The second part of the paper presents a specific approach taken to gesture recognition intended to support natural interaction with autonomous robots that guide visitors in museums and exhibition centers. The proposed gesture recognition system builds on a probabilistic framework that allows the utilization of multiple information cues to efficiently detect image regions that belong to human hands. Tracking over time is achieved by a technique that can simultaneously handle multiple hands that may move in complex trajectories, occlude each other in the field of view of the robot's camera and vary in number over time. Dependable hand tracking, combined with fingertip detection, facilitates the definition of a small, simple, intuitive hand gestures vocabulary that can be used to support robust human robot interaction. Sample experimental results presented in this paper, confirm the effectiveness and the efficiency of the proposed approach, meeting the robustness and performance requirements of this particular case of human-computer interaction.

2 Computer Vision Techniques for Hand Gesture Recognition

Most of the complete hand interactive systems can be considered to be comprised of three layers: detection, tracking and recognition. The detection layer is responsible for defining and extracting visual features that can be attributed to the presence of hands in the field of view of the camera(s). The tracking layer is responsible for performing temporal data association between successive image frames, so that, at each moment in time, the system may be aware of "what is where". Moreover, in model-based methods, tracking also provides a way to maintain estimates of model parameters, variables and features that are not directly observable at a certain moment in time. Last, the recognition layer is responsible for grouping the spatiotemporal data extracted in the previous layers and assigning the resulting groups with labels associated to particular classes of gestures. In this section, research on these three identified subproblems of vision-based gesture recognition is reviewed.

2.1 Detection

The primary step in gesture recognition systems is the detection of hands and the segmentation of the corresponding image regions. This segmentation is crucial because it isolates the task-relevant data from the image background,

before passing them to the subsequent tracking and recognition stages. A large number of methods have been proposed in the literature that utilize a several types of visual features and, in many cases, their combination. Such features are skin color, shape, motion and anatomical models of hands. In [CPC06], a comparative study on the performance of some hand segmentation techniques can be found.

2.1.1 Color

Skin color segmentation has been utilized by several approaches for hand detection. A major decision towards providing a model of skin color is the selection of the color space to be employed. Several color spaces have been proposed including RGB, normalized RGB, HSV, YCrCb, YUV, etc. Color spaces efficiently separating the chromaticity from the luminance components of color are typically considered preferable. This is due to the fact that by employing chromaticity-dependent components of color only, some degree of robustness to illumination changes can be achieved. Terrillon et al [TSFA00] review different skin chromaticity models and evaluate their performance.

To increase invariance against illumination variability some methods [MC97, Bra98, Kam98, FM99, HVD+99a, KOKS01] operate in the HSV [SF96], YCrCb [CN98], or YUV [YLW98, AL04b] colorspaces, in order to approximate the "chromaticity" of skin (or, in essence, its absorption spectrum) rather than its apparent color value. They typically eliminate the luminance component, to remove the effect of shadows, illumination changes, as well as modulations of orientation of the skin surface relative to the light source(s). The remaining 2D color vector is nearly constant for skin regions and a 2D histogram of the pixels from a region containing skin shows a strong peak at the skin color. Regions where this probability is above a threshold are detected and described using connected components analysis. In several cases (e.g. [AL04b]), hysteresis thresholding on the derived probabilities is also employed prior to connected components labeling. The rationale of hysteresis thresholding is that pixels with relatively low probability of being skin-colored, should be interpreted as such in case that they are connected to pixels with high such probability.Having selected a suitable color space, the simplest approach for defining what constitutes skin color is to employ bounds on the coordinates of the selected space [CN98]. These bounds are typically selected empirically, i.e. by examining the distribution of skin colors in a preselected set of images. Another approach is to assume that the probabilities of skin colors follow a distribution that can be learned either off-line or by employing an on-line iterative method [SF96].

Several methods [SF96, KK96, SWP98, DKS01, JR02, AL04b, SSA04] utilize precomputed color distributions extracted from statistical analysis of large datasets. For example, in [JR02], a statistical model of skin color was obtained from the analysis of thousands of photos on the Web. In contrast, methods such as those described in [KOKS01, ZYW00] build a color model based on collected samples of skin color during system initialization.

When using a histogram to represent a color distribution (as for example

in [JR02, KK96, WLH00]), the color space is quantized and, thus, the level of quantization affects the shape of the histogram. Parametric models of the color distribution have also been used in the form of a single Gaussian distribution [KKAK98, YLW98, CG99] or a mixture of Gaussians [RMG98, RG98, SRG99, JP97, JRP97]. Maximum-likelihood estimation techniques can be thereafter utilized to infer the parameters of the probability density functions. In another parametric approach [WLH00], an unsupervised clustering algorithm to approximate color distribution is based on a self-organizing map.

The perceived color of human skin varies greatly across human races or even between individuals of the same race. Additional variability may be introduced due to changing illumination conditions and/or camera characteristics. Therefore, color-based approaches to hand detection need to employ some means for compensating for this variability. In [YA98, SSA04], an invariant representation of skin color against changes in illumination is pursued, but still with not conclusive results. In [YLW98], an adaptation technique estimates the new parameters for the mean and covariance of the multivariate Gaussian skin color distribution, based on a linear combination of previous parameters. However, most of these methods are still sensitive to quickly changing or mixed lighting conditions. A simple color comparison scheme is employed in [DKS01], where the dominant color of a homogeneous region is tested as if occurring within a color range that corresponds to skin color variability. Other approaches [Bra98, KOKS01, MC97] consider skin color to be uniform across image space and extract the pursued regions through typical region-growing and pixel-grouping techniques. More advanced color segmentation techniques rely on histogram matching [Ahm94], or employ a simple look-up table approach [KK96, QMZ95] based on the training data for the skin and possibly its surrounding areas. In [FM99, HVD+99a], the skin color blobs are detected by a method using scan lines and a Bayesian estimation approach.

In general, color segmentation can be confused by background objects that have a color distribution similar to human skin. A way to cope with this problem is based on background subtraction [RK94, GD96]. However, background subtraction is typically based on the assumption that the camera system does not move with respect to a static background. To solve this problem, some research [UO98, BNI99], has looked into the dynamic correction of background models and/or background compensation methods.

In another approach [Ahm94], the two image blobs at which the hand appears in a stereo pair are detected based on skin color. The hands are approximated by an ellipse in each image and the axes of the ellipses are calculated. By corresponding the two pairs of axes in the two images, the orientation of the hand in 3D is computed. The method in [MHP+01, HVD+99b], also uses a stereoscopic pair to estimate the position of hands in 3D space. The binocular pair could pan and tilt and, also, the zoom and fixation distance of the cameras was software-controlled. The estimated distance and position of the hands were utilized so that the system could focus attention of the hands of the user, by rotating, zooming and fixating accordingly.

Skin color is only one of many cues to be used for to hand detection. For

example, in cases where the faces also appear in the camera field of view, further processing is required to distinguish hands from faces [WADP97, YK04, ZH05]. Thus, skin color has been utilized in combination with other cues to obtain better performance. Stereoscopic information has been utilized mainly in conjunction with the skin color cue to enhance the accuracy of hand localization. In [TVdM98], stereo is combined with skin color to optimize the robustness of tracking and in [ETK91] to cope with occlusions. In [YSA95] skin detection is combined with non-rigid motion detection and in [DWT04] skin color was used to restrict the region where motion features are to be tracked. An important research direction is, therefore, the combination multiple cues. Two such approaches are described in [ADS98, SSKM98].

2.1.2 Shape

The characteristic shape of hands has been utilized to detect them in images in multiple ways. Much information can be obtained by just extracting the contours of objects in the image. If correctly detected, the contour represents the shape of the hand and is therefore not directly dependent on viewpoint, skin color and illumination. On the other hand, the expressive power of 2D shape can be hindered by occlusions or degenerate viewpoints. In the general case, contour extraction that is based on edge detection results in a large number of edges that belong to the hands but also to irrelevant background objects. Therefore, sophisticated post-processing approaches are required to increase the reliability of such an approach. In this spirit, edges are often combined with (skin-)color and background subtraction/motion cues.

In the 2D/3D drawing systems of [Kru91, Kru93, UO97, UO98], the user's hand is directly extracted as a contour by assuming a uniform background and performing real-time edge detection in this image. Examples of the use of contours as features are found in both model [KH95] and appearance based techniques [GD96, PSH96]). In [DD91], finger and arm link candidates are selected through the clustering of the sets of parallel edges. In a more global approach [GD95], hypotheses of hand 3D models are evaluated by first synthesizing the edge image of a 3D model and comparing it against the acquired edge image.

Local topological descriptors have been used to match a model with the edges in the image. In [BMP02], the shape context descriptor is proposed, which characterizes a particular point location on the shape. This descriptor is the histogram of the relative polar coordinates of all other points. Detection is based on the assumption that corresponding points on two different shapes will ideally have a similar shape context. The descriptor has been applied to a variety of obvect recognition problems [BMP02, MM02], with limited background clutter. In [SC02], all topological combinations of four points are considered in a voting matrix and one-to-one correspondences are established using a greedy algorithm.

Background clutter is effectively dealt in [IB96b, IB98a], where particle filtering is employed to learn which curves belong to a tracked contour. This technique makes shape models more robust to background noise, but shape-based methods are better suited for tracking an object once it has been acquired. The

approach in [SSK99], utilizes as input hand images against a homogeneous and planar background. The illumination is such that the hand's shadow is cast on the background plane. By corresponding high-curvature features of the hand's silhouette and the shadows, depth cues such as vanishing points are extracted and the hand's pose is estimated.

Certain methods focus on the specific morphology of hands and attempt to detect them based on characteristic hand shape features such as fingertips. The approaches in [AL06b, Mag95, VD95, AL06b] utilize curvature as a cue to fingertip detection. Another technique that has been employed in fingertip detection is template matching. Templates can be images of fingertips [CBC95] or fingers [RK95] or generic 3D cylindrical models [DS94b]. Such pattern matching techniques can be enhanced by using additional image features, like contours [RK94]. The template-matching technique was utilized also in [CBC95, OZ97], with images of the top view of fingertips as the prototype. The pixel resulting in the highest correlation is selected as the position of the target object. Apart from being very computationally expensive, template matching can not cope with neither scaling nor rotation of the target object. This problem was addressed in [CBC95] by continuously updating the template.

In [ST05], the fingertip of the user was detected in both images of a calibrated stereo pair. In these images, the two points at which this tip appears establish a stereo correspondence, which is utilized to estimate the fingertip's position is 3D space. In turn, this position is utilized by the system to estimate the distance of the finger from the desk and, therefore, determine if the user is touching it. In [Jen99], a system is described for tracking the 3D position and orientation of a finger using several cameras. Tracking is based on combining multiple sources of information including stereo range images, color segmentation and shape information. The hand detectors in [AP96] and [BOP97] utilize nonlinear modeling and a combination of iterative and recursive estimation methods to recover 3D geometry from blob correspondences across multiple images. These correspondences were thereafter utilized to estimate the translations, and orientations of blobs in world coordinates. In [AL06a], stereoscopic information is used to provide 3D positions of hand centroids and fingertips but also to reconstruct the 3D contour of detected and tracked hands in real time. In [Yin03] stereo correspondences of multiple fingertips have been utilized to calibrate a stereo pair. In the context of fingertip detection, several heuristics have also been employed. For example, for deictic gestures it can be assumed that the finger represents the foremost point of the hand [Mag95, QMZ95]. Many other indirect approaches for the detection of fingertips have been employed, like image analysis using specially tuned Gabor kernels [MR92]. The main disadvantage in the use of fingertips as features is that they can be occluded by the rest of the hand. A solution to this occlusion problem involves the use of multiple cameras [LK95, RK94]. Other solutions are based on the estimation of the occluded fingertip positions, based on the knowledge of the 3D model of the gesture in question [SSKM98, WLH01, WTH99, RK95].

2.1.3 Learning detectors from pixel values

Significant work has been carried out on finding hands in grey level images based on their appearance and texture. In [WH00], the suitability of a number of classification methods for the purpose of view-independent hand posture recognition was investigated. Several methods [CSW95, CW96b, QZ96, TM96, TVdM98] attempt to detect hands based on hand appearances, by training classifiers on a set of image samples. The basic assumption is that hand appearance differs more among hand gestures than it differs among different people performing the same gesture. Still, automatic feature selection constitutes a major difficulty. Several papers consider the problem of feature extraction [TM96, QZ96, NR98, TVdM98] and selection [CSW95, CW96b], with limited results regarding hand detection. The work in [CW96b], investigates the difference between the most discriminating features (MDFs) and the most expressive features (MEFs) in the classification of motion clips that contain gestures. It is argued that MEFs may not be the best for classification, because the features that describe some major variations in the class are, typically, irrelevant to how the sub-classes are divided. MDFs are selected by multi-class, multivariate discriminate analysis and have a significantly higher capability to catch major differences between classes. Their experiments also showed that MDFs are superior to the MEFs in automatic feature selection for classification.

More recently, methods based on a machine learning approach called boosting have demonstrated very robust results in face and hand detection. Due to these results, they are reviewed in more detail below. Boosting is a general method that can be used for improving the accuracy of a given learning algorithm [Sch02]. It is based on the principle that a highly accurate or "strong" classifier can be derived through the linear combination of many relatively inaccurate or "weak" classifiers. In general, an individual weak classifier is required to perform only slightly better than random. As proposed in [VJ01] for the problem of hand detection, a weak classifier might be a simple detector based on basic image block differences efficiently calculated using an integral image.

The AdaBoost algorithm [FS97] provides a learning method for finding suitable collections of weak classifiers. For training, it employs an exponential loss function that models the upper bound of the training error. The method utilizes a training set of images that consists of positive and negative examples (hands and non-hands, in this case), which are associated with corresponding labels. Weak classifiers are added sequentially into an existing set of already selected weak classifiers in order to decrease the upper bound of the training error. It is known that this is possible if weak classifiers are of a particular form [FHR00, SS98]. AdaBoost was applied to the area of face and pedestrian detection [VJ01, VJS03] with impressive results. However, this method may result in an excessive number of weak classifiers. The problem is that AdaBoost does not consider the removal of selected weak classifiers that no longer contribute to the detection process. The FloatBoost algorithm proposed in [LZ04] extends the original AdaBoost algorithm, in that it removes an existing weak classifier from a strong classifier if it no longer contributes to the decrease of the training error.

This results in a more general therefore more efficient set of weak classifiers.

In the same context, the final detector can be divided into a cascade of strong classifier layers [VJ01]. This hierarchical structure is comprised of a general detector at the root, with branch nodes being increasingly more appearance-specific as the depth of the tree increases. In this approach, the larger the depth of a nodes the more specific the training set becomes. To create a labeled database of training images for the above tree structure, an automatic method [OB04] for performing grouping of images of hands at the same posture is proposed, based on an unsupervised clustering technique.

2.1.4 3D model-based detection

A category of approaches utilize 3D hand models for the detection of hands in images. One of the advantages of these methods is that they can achieve view-independent detection. The employed 3D models should have enough degrees of freedom to adapt to the dimensions of the hand(s) present in an image.

Different models require different image features to construct feature-model correspondences. Point and line features are employed in kinematic hand models to recover angles formed at the joints of the hand [RK95, SSKM98, WTH99, WLH01]. Hand postures are then estimated provided that the correspondences between the 3D model and the observed image features are well established. Various 3D hand models have been proposed in the literature. In [RK94, SMC02], a full hand model is proposed which has 27 degrees of freedom (DOF) (6 DOF for 3D location/orientation and 21 DOF for articulation). In [LWH02], a "cardboard model" is utilized, where each finger is represented by a set of three connected planar patches. In [GdBUP95], a 3D model of the arm with 7 parameters is utilized. In [GD96], a 3D model with 22 degrees of freedom for the whole body with 4 degrees of freedom for each arm is proposed. In [MI00], the user's hand is modeled much more simply, as an articulated rigid object with three joints comprised by the first index finger and thumb.

In [RK94], edge features in the two images of a stereoscopic pair are corresponded to extract the orientation of in-between joints of fingers. These are subsequently utilized for model based tracking of the hands. In [NR98], artificial neural networks that are trained with body landmarks, are utilized for the detection of hands in images. Some approaches [HH96b, HH96a, LK95] utilize a deformable model framework to fit a 3D model of the hand to image data. The fitting is guided by forces that attract the model to the image edges, balanced by other forces that tend to preserve continuity and evenness among surface points [HH96b, HH96a]. In [LK95], the process is enhanced with anatomical data of the human hand that are incorporated into the model. Also, to fit the hand model to an image of a real hand, characteristic points on the hand are identified in the images, and virtual springs are implied which pull these characteristic points to goal positions on the hand model.

2.1.5 Motion

Motion is a cue utilized by a few approaches to hand detection. The reason is that motion-based hand detection demands for a very controlled setup, since it assumes that the only motion in the image is due to hand movement. Indeed, early works (e.g. [FW95, Que95, CW96b]) assumed that hand motion is the only motion occurring in the imaged environment. In more recent approaches, motion information is combined with additional visual cues. In the case of static cameras, the problem of motion estimation reduces to that of background maintenance and subsequent subtraction. For example in [CT98, MDC98] such information is utilized to distinguish hands from other skin-colored objects and cope with lighting conditions imposed by colored lights. The difference in luminance of pixels from two successive images is close to zero for pixels of the background. By choosing and maintaining an appropriate threshold, moving objects are detected within a static scene.

In [YSA95], a novel feature, based on motion residue, is proposed. Hands typically undergo non-rigid motion, because they are articulated objects. Consequently, hand detection capitalizes on the observation that for hands, inter-frame appearance changes are more frequent than for other objects such as clothes, face, and background.

2.2 Tracking

Tracking, or the frame-to-frame correspondence of the segmented hand regions or features, is the second step in the process towards understanding the observed hand movements. The importance of robust tracking is twofold. First, it provides the inter-frame linking of hand/finger appearances, giving rise to trajectories of features in time. These trajectories convey essential information regarding the gesture and might be used either in a raw form (e.g. in certain control applications like virtual drawing the tracked hand trajectory directly guides the drawing operation) or after further analysis (e.g. recognition of a certain type of hand gesture). Second, in model-based methods, tracking also provides a way to maintain estimates of model parameters variables and features that are not directly observable at a certain moment in time.

2.2.1 Template based tracking

This class of methods exhibits great similarity to methods for hand detection. Members of this class invoke the hand detector at the spatial vicinity that the hand was detected in the previous frame, so as to drastically restrict the image search space. The implicit assumption for this method to succeed is that images are acquired frequently enough.

Correlation-based feature tracking is directly derived from the above approach. In [CBC95, OZ97] correlation-based template matching is utilized to track hand features across frames. Once the hand(s) have been detected in a frame, the image regions in which they appear is utilized as the prototype to

detect the hand in the next frame. Again, the assumption is that hands will appear in the same spatial neighborhood. This technique is employed for a static camera in [DEP96], to obtain characteristic patterns (or "signatures") of gestures, as seen from a particular view. The work in [HB96] deals also with variable illumination. A target is viewed under various lighting conditions. Then, a set of basis images that can be used to approximate the appearance of the object viewed under various illumination conditions is constructed. Tracking simultaneously solves for the affine motion of the object and the illumination. Real-time performance is achieved by pre-computing "motion templates" which are the product of the spatial derivatives of the reference image to be tracked and a set of motion fields.

Some approaches detect hands as image blobs in each frame and temporally correspond blobs that occur in proximate locations across frames. Approaches that utilize this type of blob tracking are mainly the ones that detect hands based on skin color, the blob being the correspondingly segmented image region (e.g. [BMM97, AL04b]). Blob-based approaches are able to retain tracking of hands even when there are great variations from frame to frame.

Extending the above approach, deformable contours, or "snakes" have been utilized to track hand regions in successive image frames [CJ92]. Typically, the boundary of this region is determined by intensity or color gradient. Nevertheless, other types of image features (e.g. texture) can be considered. The technique is initialized by placing a contour near the region of interest. The contour is then iteratively deformed towards nearby edges to better fit the actual hand region. This deformation is performed through the optimization of an "energy" functional that sums up the gradient at the locations of the snake while, at the same time, favoring the smoothness of the contour. When snakes are used for tracking, an active shape model is applied to each frame and the convergence of the snake in that frame is used as a starting point for the next frame. Snakes allow for real-time tracking and can handle multiple targets as well as complex hand postures. They exhibit better performance when there is sufficient contrast between the background and the object [CJHG95]. On the contrary, their performance is compromised in cluttered backgrounds. The reason is that the snake algorithm is sensitive to local optima of the energy function, often due to ill foreground/background separation or large object diplacements and/or shape deformations between successive images.

Tracking local hand features on the hand has been employed in specific contexts only, probably because tracking local features does not guarantee the segmentation of the hands from the rest of the image. The methods in [MDC98, BH94], track hands in image sequences by combining two motion estimation processes, both based on image differencing. The first process computes differences between successive images. The second computes differences from a background image that was previously acquired. The purpose of this combination is increased robustness near shadows.

2.2.2 Optimal estimation techniques

Feature tracking has been extensively studied in computer vision. In this context, the optimal estimation framework provided by the Kalman filter [Kal60] has been widely employed in turning observations (feature detection) into estimations (extracted trajectory). The reasons for its popularity are real-time performance, treatment of uncertainty, and the provision of predictions for the successive frames.

In [AL04b], the target is retained against cases where hands occlude each other, or appear as a single blob in the image, based on a hypothesis formulation and validation/rejection scheme. The problem of multiple blob tracking was investigated in [AL04a], where blob tracking is performed in both images of a stereo pair and blobs are corresponded, not only across frames, but also across cameras. The obtained stereo information not only provides the 3D locations of the hands, but also facilitates the potential motion of the observing stereo pair which could be thus mounted on a robot that follows the user. In [BK98, Koh97], the orientation of the user's hand was continuously estimated with the Kalman filter to localize the point in space that the user indicates by extending the arm and pointing with the index finger. In [UO99], hands are tracked from multiple cameras, with a Kalman filter in each image, to estimate the 3D hand postures. Snakes integrated with the Kalman filtering framework (see below) have been used for tracking hands [DS92]. Robustness against background clutter is achieved in [Pet99], where the conventional image gradient is combined with optical flow to separate the foreground from the background. In order to provide accurate initialization for the snake in the next frame, the work in [KL01], utilizes the optical flow to obtain estimations of the direction and magnitude of the target's motion. The success of combining optical flow is based on the accuracy of its computation and, thus, the approach is best suited for the case of static cameras.

Treating the tracking of image features within a Bayesian framework has been long known to provide improved estimation results. The works in [FB02, IB98b, VPGB02, HLCP02, IM01, KMA01] investigate the topic within the context of hand and body motion. In [WADP97], a system tracks a single person by color-segmentation of the image into blobs and then uses prior information about skin color and topology of a person's body to interpret the set of blobs as a human figure. In [Bre97], a method is proposed for tracking human motion by grouping pixels into blobs based on coherent motion, color and temporal support using an expectation-maximization (EM) algorithm. Each blob is subsequently tracked using a Kalman filter. Finally, in [MB99, MI00], the contours of blobs are tracked across frames by a combination of the Iterative Closed Point (ICP) algorithm and a factorization method to determine global hand pose.

The approaches in [BJ96, BJ98c] reformulate the eigenspace reconstruction problem (reviewed in Section 2.3.2) as a problem of robust estimation. The goal is to utilize the above framework to track the gestures of a moving hand. To account for large affine transformations between the eigenspace and the image, a multi-scale eigenspace representation is defined and a coarse-to-fine matching

strategy is adopted. In [LB96], a similar approach was proposed which uses a hypothesize-and-test approach instead of a continuous formulation. Although this approach does not address parameterized transformations and tracking, it exhibits robustness against occlusions. In [GMR+02], a real-time extension of the work in [BJ96], based on EigenTracking [IB98a] is proposed. Eigenspace representations have been utilized in a different way in [BH94] to track articulated objects by tracking a silhouette of the object, which was obtained via image differencing. A spline was fit to the object's outline and the knot points of the spline form the representation of the current view. Tracking an object amounts to projecting the knot points of a particular view onto the eigenspace. Thus, this work uses the shape (silhouette) information instead of the photometric one (image intensity values).

In [UO99], the 3D positions and postures of both hands are tracked using multiple cameras. Each hand position is tracked with a Kalman filter and 3D hand postures are estimated using image features. This work deals with the mutual hand-to-hand occlusion inherent in tracking both hands, by selecting the views in which there are no such occlusions.

2.2.3 Tracking based on the Mean Shift algorithm

The Mean Shift algorithm [Che95] is an iterative procedure that detects local maxima of a density function by shifting a kernel towards the average of data points in its neighborhood. The algorithm is significantly faster than exhaustive search, but requires appropriate initialization.

The Mean Shift algorithm has been utilized in the tracking of moving objects in image sequences. The work in [CRM00, CRM03] is not restricted to hand tracking, but can be used to track any moving object. It characterizes the object of interest through its color distribution as this appears in the acquired image sequence and utilizes the spatial gradient of the statistical measurement towards the most similar (in terms of color distribution similarity) image region. An improvement of the above approach is described in [CL01], where the mean shift kernel is generalized with the notion of the "trust region". Contrary to mean shift which directly adopts the direction towards the mean, trust regions attempt to approximate the objective function and, thus, exhibit increased robustness towards being trapped in spurious local optima. In [Bra98], a version of the Mean Shift algorithm is utilized to track the skin-colored blob of a human hand. For increased robustness, the method tracks the centroid of the blob and also continuously adapts the representation of the tracked color distribution. Similar is also the method proposed in [KOKS01], except the fact that it utilizes a Gaussian mixture model to approximate the color histogram and the EM algorithm to classify skin pixels based on the Bayesian decision theory.

Mean-Shift tracking is robust and versatile for a modest computational cost. It is well suited for tracking tasks where the spatial structure of the tracked objects exhibits such a great variability that trackers based on a space-dependent appearance reference would break down very fast. On the other hand, highly cluttered background and occlusions may distract the mean-shift trackers from

the object of interest. The reason appears to be its local scope in combination with the single-state appearance description of the target.

2.2.4 Particle filtering

Particle filters have been utilized to track the position of hands and the configuration of fingers in dense visual clutter. In this approach, the belief of the system regarding the location of a hand is modeled with a set of particles. The approach exhibits advantages over Kalman filtering, because it is not limited by the unimodal nature of Gaussian densities that cannot represent simultaneous alternative hypotheses. A disadvantage of particle filters is that for complex models (such as the human hand) many particles are required, a fact which makes the problem intractable especially for high-dimensional models. Therefore, other assumptions are often utilized to reduce the number of particles. For example in [IB98a], dimensionality is reduced by modeling commonly known constraints due to the anatomy of the hand. Additionally, motion capture data are integrated in the model. In [MB99] a simplified and application-specific model of the human hand is utilized.

The CONDENSATION algorithm [IB98a] which has been used to learn to track curves against cluttered backgrounds, exhibits better performance than Kalman filters, and operates in real-time. It uses "factored sampling", previously applied to the interpretation of static images, in which the probability distribution of possible interpretations is represented by a randomly generated set. Condensation uses learned dynamical models, together with visual observations, to propagate this random set over time. The result is highly robust tracking of agile motion. In [MI00] the "partitioned sampling" technique is employed to avoid the high computational cost that particle filters exhibit when tracking more than one object. In [LL01], the state space is limited to 2D translation, planar rotation, scaling and the number of outstretched fingers.

Extending the CONDENSATION algorithm the work in [MCA01], detects occlusions with some uncertainty. In [PHVG02], the same algorithm is integrated with color information; the approach is based on the principle of color histogram distance, but within a probabilistic framework, the work in introduces a new Monte Carlo tracking technique. In general, contour tracking techniques, typically, allow only a small subset of possible movements to maintain continuous deformation of contours. This limitation was overcome to some extent in [HH96b], who describe an adaptation of the CONDENSATION algorithm for tracking across discontinuities in contour shapes.

2.3 Recognition

The overall goal of hand gesture recognition is the interpretation of the semantics that the hand(s) location, posture, or gesture conveys. Basically, there have been two types of interaction in which hands are employed in the user's communication with a computer. The first is control applications such as drawing, where the user sketches a curve while the computer renders this curve on a 2D

canvas [LWH02, WLH01]. Methods that relate to hand-driven control focus on the detection and tracking of some feature (e.g. the fingertip, the centroid of the hand in the image etc) and can be handled with the information extracted through the tracking of these features. The second type of interaction involves the recognition of hand postures, or signs, and gestures. Naturally, the vocabulary of signs or gestures is largely application dependent. Typically, the larger the vocabulary is, the hardest the recognition task becomes. Two early systems indicate the difference between recognition [BMM97] and control [MM95]. The first recognizes 25 postures from the International Hand Alphabet, while the second was used to support interaction in a virtual workspace.

The recognition of postures is of topic of great interest on its own, because of sign language communication. Moreover, it also forms the basis of numerous gesture-recognition methods that treat gestures as a series of hand postures. Besides the recognition of hand postures from images, recognition of gestures includes an additional level of complexity, which involves the parsing, or segmentation, of the continuous signal into constituent elements. In a wide variety of methods (e.g. [TVdM98]), the temporal instances at which hand velocity (or optical flow) is minimized are considered as observed postures, while video frames that portray a hand in motion are sometimes disregarded (e.g. [BMM97]). However, the problem of simultaneous segmentation and recognition of gestures without being confused with inter-gesture hand motions remains a rather challenging one. Another requirement for this segmentation process is to cope with the shape and time variability that the same gesture may exhibit, e.g. when performed by different persons or by the same person at different speeds.

The fact that even hand posture recognition exhibits considerable levels of uncertainty casts the above processing computationally complex or error prone. Several of the reviewed works indicate that lack of robustness in gesture recognition can be compensated by addressing the temporal context of detected gestures. This can be established by letting the gesture detector know of the grammatical or physical rules that the observed gestures are supposed to express. Based on these rules, certain candidate gestures may be improbable. In turn, this information may disambiguate candidate gestures, by selecting to recognize the most likely candidate. The framework of Hidden Markov Models (HMMs) that is discussed later in this section, provides a suitable framework for modeling the context-dependent reasoning of the observed gestures.

2.3.1 Template matching

Template matching, a fundamental pattern recognition technique, has been utilized in the context of both posture and gesture recognition. In the context of images, template matching is performed by the pixel-by-pixel comparison of a prototype and a candidate image. The similarity of the candidate to the prototype is proportional to the total score on a preselected similarity measure. For the recognition of hand postures, the image of a detected hand forms the candidate image which is directly compared with prototype images of hand postures. The best matching prototype (if any) is considered as the matching posture.

Clearly, because of the pixel-by-pixel image comparison, template matching is not invariant to scaling and rotation.

Template matching was one of the first methods employed to detect hands in images [FW95]. To cope with the variability due to scale and rotation, some authors have proposed scale and rotational normalization methods (e.g. [BMM97]), while others equip the set of prototypes with images from multiple views (e.g. [DP93]). In [BMM97], the image of the hand is normalized for rotation based on the detection of the hands main axis and, then, scaled with respect to hand dimensions in the image. Therefore, in this method the hand is constrained to move on a planar surface that is frontoparallel to the camera. To cope with the increased computational cost when comparing with multiple views of the same prototype, these views were annotated with the orientation parameters [FAK03]. Searching for the matching prototype was accelerated, by searching only in relevant postures with respect to the one detected in the previous frame. A template comprised of edge directions was utilized in [FR95]. Edge detection is performed on the image of the isolated hand and edge orientations are computed. The histogram of these orientations is used as the feature vector. The evaluation of this approach showed that edge orientation histograms are not very discriminative, because several semantically different gestures exhibit similar histograms.

A direct approach of including the temporal component into the template matching techniques has been proposed in [DP93, DP95, DEP96]. For each input frame, the (normalized) hand image region is compared to different views of the same posture and a 1D function of responses for each posture is obtained; due to the dense posture parameterization this function exhibits some continuity. By stacking the 1D functions resulting from a series of input frames, a 2D pattern is obtained and utilized as a template.

Another approach to creating gesture patterns that can be matched by templates, is to accumulate the motion over time within a "motion" or "history" image. The input images are processed frame-by-frame and some motion-related feature is detected at each frame. The detected features, from all frames, are accumulated in a 2D buffer at the location of their detection. The obtained image is utilized as a representation of the gesture and serves as a recognition pattern. By doing so, the motion (or trail) of characteristic image points over the sequence is captured. The approach is suited for a static camera observing a single user in front of a static background. Several variations of this basic idea have been proposed. In [BD96, BD01] the results of a background subtraction process (human silhouettes) are accumulated in a single frame and the result is utilized as the feature vector. In [Dav01, BD00, BD02], an extension of the previous idea encodes temporal order in the feature vector, by creating a "history gradient". In the accumulator image, older images are associated with a smaller accumulation value and, so, they form a "fading-out" pattern. Similar is the approach in [CT98], where the accumulation pattern is comprised of optical flow vectors. The obtained pattern is rather coarse, but with the use of a user-defined rule-based technique the system can distinguish only among a very small vocabulary of coarse body gestures. In [YAT02], an artificial neural

network is trained to learn motion patterns similar to the above. In [ICLB05], a single hand was imaged in a control environment that featured no depth and its (image) skeleton was computed; the accumulation of such skeletons along time, in a single image, was used as the feature vector.

2.3.2 Methods based on Principal Component Analysis

Principal Component Analysis (PCA) methods have been directly utilized mainly in posture recognition. However, this analysis facilitates several gesture recognition systems by providing the "tokens" to be used as input to recognition.

PCA methods require an initial training stage, in which a set of images of similar content is processed. Typically, the intensity values of each image are considered as values of a 1D vector, whose dimensionality is equal to the number of pixels in the image; it is assumed, or enforced, that all images are of equal size. For each such set, some basis vectors are constructed that can be used to approximate any of the (training) images in the set. In the case of gesture recognition, the training set contains images of hands in certain postures. The above process is performed for each posture in the vocabulary, which the system should later be able to recognize. In PCA-based gesture recognition, the matching combination of principal components indicates the matching gesture as well. This is because the matching combination is one of the representatives of the set of gestures that were clustered together in training, as expressions of the same gesture. A problem of eigenspace reconstruction methods is that they are not invariant to image transformations such as translation, scaling, and rotation.

PCA was first applied to recognition in [SK87]and later extended in [TP91] and [MN95]. A simple system is presented in [RA97], where the whole image of a person gesturing is processed, assuming that the main component of motion is the gesture. View-dependency is compensated by creating multiple prototypes, one for each view. As in detection, the matching view indicates also the relative pose to the camera. To reduce this complexity in recognition, the system in [BMM97], rotationally aligns the acquired image with the template based on the arm's orientation and, therefore, stores each gesture prototype in a single orientation.

PCA systems exhibit the potential capability of compressing the knowledge of the system by keeping only the principal components with the n highest eigenvalues. However, in [BMM97], it was shown that this is not effective if only a small number of principal components are to be kept. The works in [CH96, MP95] attempt to select the features that best represent the pattern class, using an entropy based analysis. In a similar spirit, in [CSW95, CW96b] features that better represent a class (expressive) are compared to features that maximize the dissimilarity across classes (discriminative), to suggest that the latter give rise to more accurate recognition results. A remarkable extension of this work is the utilization of the recognition procedure as feedback to the hand segmentation process [CW96a]. In that respect, authors utilize the classification procedure in combination with hand detection to eliminate unlikely segmentations.

2.3.3 Boosting

The learning methods reviewed in section 2.1.3 have remarkable performance in hand detection and hand posture recognition, but limited application in hand gesture recognition. Here, characteristic examples of the use of these methods for posture recognition are reviewed.

In [LF02], a real-time gesture recognition system is presented. Their method which is based on skin-color segmentation, is facilitated by a boosting algorithm [FS97] for fast classification. To normalize for orientation, the user is required to wear a wristband so that the hand shape can be easily mapped to a canonical frame. In [TPS03], a classification approach was proposed, together with parameter interpolation to track hand motion. Image intensity data was used to train a hierarchical nearest neighbor classifier, classifying each frame as one of 360 views, to cope with viewpoint variability. This method can handle fast hand motion, but it relies on clear skin color segmentation and controlled lighting conditions. In [WKSE02], the hand is detected and the corresponding image segment is subsampled to a very low resolution. The pixels of the resulting patterns are then treated as N-dimensional vectors. Learning in this case is based on a C-means clustering of the training parameter space.

2.3.4 Contour and silhouette matching

This class of methods mainly refers to posture recognition and is conceptually related to template matching in that it compares prototype images with the hand image that was acquired to obtain a match. The defined feature space is the edges of the above images. The fact that a spatially sparser feature is utilized (edges instead of intensities) gives rise to the employment of slightly different similarity metrics in the comparison of acquired and prototype images. In addition, continuity is favored in order to avoid the consideration of spurious edges that may belong e.g. to background clutter.

In [Bor88] and [GD96], Chamfer matching [BTW77] is utilized as the similarity metric. In [SMC02], matching is based on an "Unscented Kalman filter", which minimizes the geometric error between the profiles and edges extracted from the images. The same, edge, image features are utilized in [DBR00] and recognition is completed after a likelihood analysis. The work in [Bor88] applies a coarse-to-fine search, based on a resolution pyramid of the image, to accelerate the process. In an image-generative approach [GD96], the edges of idealized models of body postures are projected onto images acquired from multiple views and compared with the true edges using Chamfer matching while, also, a template hierarchy is used to handle shape variation. In [OH97], a template hierarchy is also utilized to recognize 3D objects from different views and the Hausdorff distance [HKR93] is utilized as the similarity metric. In a more recent approach, the work in [CKBH00] utilizes the robust "shape context" [BMP02] matching operator.

The research in [RASS01, AS01, AS02, AS03] utilizes Chamfer matching between input and model edge images. The model images are a priori synthe-

sized with the use of a data-glove. The number of model images is very high ($\approx 10^5$) in order to capture even minute differences in postures. To cope with this amount of data, the retrieval is performed hierarchically, by first rejecting the greatest proportion of all database views, and then ranking the remaining candidates in order of similarity to the input. In [AS02], the Chamfer matching technique was evaluated against edge orientation histograms, shape moments and detected finger positions.

The use of silhouettes in gesture recognition has not been extensive, probably because different hand poses can give rise to the same or similar silhouette. Another reason is that silhouette matching requires alignment (or else, point-to-point correspondence establishment across the total arclength), which is not always a trivial task. Also, matching of silhouettes using their conventional arclength descriptions (or "signatures") is very sensitive to deformations and noise. Due to the local nature of edges, perceptually small dissimilarities of the acquired silhouette with the prototype may cause large metric dissimilarity. Thus, depending on the metric, the overall process can be sensitive even to small shape variations, which are due to hand articulation in-between stored poses of the hand. To provide some flexibility against such variations, the work in [STTC06] aligns the contours to be matched using the Iterative Closest Point (ICP) algorithm [BM92]. A more effective approach for dealing with this variability is presented in [SSK99], where the intrusions and protrusions of the hand's silhouette are utilized as classification features.

In [LTA95], a simple contour matching technique was proposed that targeted posture recognition. In [KH95], contour matching is enabled mainly for control and coarse hand modeling. The approach in [HS95], employs a silhouette alignment and matching technique to recognize a prototype hand-silhouette in an image and subsequently track it. Finally, polar-coordinate descriptions of the contours points, or "signatures" [BF95]) and "size functions" [UV95] have been used. Similar is also the approach in [SKS01] which, after extracting the silhouette of a hand, it computes a silhouette-based descriptor that the recognition will be based upon. Because this descriptor is a function of the contour's arclength, it is very sensitive to deformations that alter the circumference of the contour and, thus, the authors propose a compensation technique. In addition, to reduce the search space of each recognition query, an adjacency map indexes the database of models. In each frame, the search space is limited to the "adjacent" views of the one estimated in the previous frame.

2.3.5 Model-based recognition methods

Most of the model-based gesture recognition approaches employ successive approximation methods for the estimation of their parameters. Since gesture recognition is required to be invariant of relative rotation, intrinsic parameters such as joint angles are widely utilized. The strategy of most methods in this category is to estimate the model parameters, e.g. by inference or optimization, so that the extracted features match a model.

In an early approach [Que95], the 3D trajectory of hands was estimated in

the image, based on optical flow. The extremal points of the trajectory were detected and used as gesture classification features. In [CBA+96a], the 3D trajectories of hands are acquired by stereo vision and utilized for HMM-based learning and recognition of gestures. Different feature vectors were evaluated as to their efficacy in gesture recognition. The results indicated that choosing the right set of features is crucial to the obtained performance. In particular, is was observed that velocity features are superior to positional features, while partial rotational invariance is also a discriminative feature.

In [DS94a], a small vocabulary of gestures are recognized through the projection of fingertips on the image plane. Although the detection is based on markers, a framework is offered that uses only the fingertips as input data and permits a model that represents each fingertip trajectory through space as a simple vector. The model is simplified in that it assumes that most finger movements are linear and exhibit minute rotational motion. Also in [KI93], grasps are recognized after estimating finger trajectories from both passive and active vision techniques [KI91]. However, the authors formulate the grasp-gesture detector in the domain of 3D trajectories, offering, at the same time, a detailed modeling of grasp kinematics (see [KI91] for a review on this topic).

The approach in [BW97], uses a "time-collapsing" technique for computing a prototype trajectory of an ensemble of trajectories, in order to extract prototypes and recognize gestures from an unsegmented, continuous stream of sensor data. The prototype offers a convenient arclength parameterization of the data points, which is then used to calculate a sequence of states along the prototype. A gesture is defined as an ordered sequence of states along the prototype and the feature space is divided into a relatively small number of finite states. A particular gesture is recognized as a sequence of transitions through a series of such states thus casting a relationship to HMM-based approaches (see Section 2.3.6). In [KM03], continuous states are utilized for gesture recognition in a multi-view context. In [CBA+96b], the 3D trajectories of hands when gesturing are estimated based on stereoscopic information and, in turn, features of these trajectories, such as orientation, velocity etc, are estimated. Similarly, for the purpose of studying of two-handed movements, the method in [SS05] estimates features of 3D gesture trajectories.

In [WP97], properties such as blob trajectories are encoded in 1D functions of time and then matched with gesture patterns using dynamic temporal warping (DTW). In [EGG+03], a framework is presented for the definition of templates encoding the motion and posture of hands using predicates that describe the postures of fingers at a semantic level. Such data-structures are considered to be semantic representations of gestures and are recognized, via template-matching, as certain gesture prototypes.

The approach presented in [BJ98a, BJ98b] achieves the recognition of gestures given some estimated representation of the hand motion. For each pair of frames in a video sequence, a set of parameters that describe the motion are computed, such as velocity or optical flow. These parameter vectors form temporal trajectories that characterize the gesture. For a new image sequence, recognition is performed by incrementally matching the estimated trajectory to

the prototype ones. Robust tracking of the parameters is based on the CONDENSATION tracking algorithm [IB96b, IB96a]. The work in [IB98c] is also similar to the above, showing that the CONDENSATION algorithm is compatible with simple dynamical models of gestures to simultaneously perform tracking and recognition. The work in [GWP99], extends the above approach in including HMMs to increase recognition accuracy.

In [LWH02], the hand gesture is estimated by matching the 3D model projections and observed image features, so that the problem becomes a search problem in a high dimensional space. In such approaches, tracking and recognition are tightly coupled: since by detecting or tracking the hand the gesture is already recognized. For this reason, these methods are discussed in more depth in section 2.3. In [RASS01], the low level visual features of hand joint configuration were mapped with a supervised learning framework for training the mapping function. In [WH00] the supervised and the unsupervised learning framework was combined and, thus, incorporate a large set of unlabeled training data. The major advantage of using appearance based methods is the simplicity of their parameter computation. However, the mapping may not be one-to-one, and the loss of precise spatial information makes them especially less suited for hand position reconstruction.

2.3.6 HMMs

A Hidden Markov Model (HMM) is a statistical model in which a set of hidden parameters is determined from a set of related, observable parameters. In a HMM, the state is not directly observable, but instead, variables influenced by the state are. Each state has a probability distribution over the possible output tokens. Therefore, the sequence of tokens generated by an HMM provides information about the sequence of states. In the context of gesture recognition, the observable parameters are estimated by recognizing postures (tokens) in images. For this reason and because gestures can be recognized as a sequence of postures, HMMs have been widely utilized for gesture recognition. In this context, it is typical that each gesture is handled by a different HMM. The recognition problem is transformed to the problem of selecting the HMM that matches best the observed data, given the possibility of a state being observed with respect to context. This context may be spelling or grammar rules, the previous gestures, cross-modal information (e.g. audio) and others. An excellent introduction and further analysis on the approach, for the case of gesture recognition, can be found in [WB95].

Early versions of this approach can be found in [YOI92, SHJ94a, RKS96]. There, the the HMMs were performing directly on the intensity values of the images acquired by a static camera. In [ML97], the edge image combined with intensity information is used to create a static posture representation or a search pattern. The work in [RKE98] includes the temporal component in an approach similar to that of [BD96] and HMMs are trained on a 2D “motion image”. The method operates on coarse body motions and visually distinct gestures executed on a plane that is frontoparallel to the camera. Images are acquired in

a controlled setting, where image differencing is utilized to construct the required motion image. Incremental improvements of this work have been reported in [EKR+98].

The work in [VM98], proposes a posture recognition system whose inputs are 3D reconstructions of the hand (and body) articulation. In this work, HMMs are coupled with 3D reconstruction methods to increase robustness. In particular, moving limbs are extracted from images, using the segmentation of [KMB94] and, subsequently, joint locations are recovered by inferring the articulated motion from the silhouettes of segments. The process is performed simultaneously from multiple views and the stereo combination of these segmentations provides the 3D models of these limbs which are, in turn, utilized for recognition.

In [SWP98], the utilized features are the moments of skin-color based blob extraction for two observed hands. Grammar rules are integrated in the HMM to increase robustness in the comprehension of gestures. This way, posture-combinations can be characterized as erroneous or improbable depending on previous gestures. In turn, this information can be utilized as feedback to increase the robustness of the posture recognition task and, thus, produce overall more accurate recognition results. The approach in [LK99], introduces the concept of a threshold model that calculates the likelihood threshold of an input (moments of blob detection). The threshold model is a weak model for the superset of all gestures in the vocabulary and its likelihood is smaller than that of the correct gesture model for a given gesture, but larger than for a non-gesture motion. This can be utilized to detect if some motion is part of a gesture or not. To reduce the states model, states with similar probability distributions are merged, based on a relative entropy measure. In [WP97], the 3D locations that result from stereo multiple-blob tracking are input to a HMM that integrates a skeletal model of the human body. Based on the 3D observations, the approach attempts to infer the posture of the body.

Conceptually similar to conditional based reasoning is the "causal analysis" approach. This approach stems from work in scene analysis [BBC93], which was developed for rigid objects of simple shape (blocks, cubes etc). The approach uses knowledge about body kinematics and dynamics to identify gestures based on human motor plans, based on measurements of shoulder, elbow and wrist joint positions in the image plane. From these positions, the system extracts a feature set that includes wrist acceleration and deceleration, effort to lift the hand against gravity, size of gesture, area between arms, angle between forearms, nearness to body etc. Gesture filters use this information, along with causal knowledge on humans interaction with objects in the physical world, to recognize gestures such as opening, lifting, patting, pushing, stopping, and clutching.

2.4 Complete gesture recognition systems

Systems that employ hand driven human-computer communication, interpret the actions of hands in different modes of interaction depending on the application domain. In some applications the hand or finger motion is tracked to be

replicated in some kind of 2D or 3D manipulation activity. For example, in a painting application the finger may sketch a figure in thin air, which however, is to be replicated as a drawing on the computer's screen. In other cases, the posture, motion, and/or gesture of the user must be interpreted as a specific command to be executed or a message to be communicated. Such a specific application domain is sign language understanding for the hearing impaired. Most of the systems presented in this subsection fall in these categories, however, there are some that combine the above two modes of interaction. Finally, a few other applications focus on gesture recognition for understanding and annotating human behavior, while others attempt to model hand and body motion for physical training.

The use of a pointing finger instead of the mouse cursor appears to be an intuitive choice in hand-driver interaction, as it has been adopted by a number of systems - possibly due to the cross-culture nature of the gesture as well as its straightforward detection in the image. In [FSM94], a generic interface that estimates the location and orientation of the pointing finger was introduced. In [CBC95], the motion of the user's pointing finger indicates the line of drawing in a "FingerPaint" application. In [Que96], 2D finger movements are interpreted as computer mouse motion in a "FingerMouse" application. In [Ahm94], the 3D position and planar orientation of the hand are tracked to provide of an interface for navigation around virtual worlds. In [WHSSdVL04], tracking of a human finger from a monocular sequence of images is performed to implement a 3D blackboard application; to recover the third dimension from the two-dimensional images the fact that the motion of the human arm is highly constrained is utilized.

The Digital Desk Calculator application [Wel93], tracked the user's pointing finger to recognize numbers on physical documents on a desk and recognize them in order to do calculations with them. The system in [SHWP07] utilizes the direction of the pointing gesture of the user to infer the object that the user is pointing at, on his/hers desk. In [KF94], a "responsive workbench" allows the user to manipulate objects in a virtual environment for industrial training via tracking of the user's hands. More recently the, very interesting, system in [BHWS04, BHW+05] attempts to recognize actions performed by the user's hands in an unconstrained office environment. Besides the fact that it applies attention mechanisms, visual learning and contextual as well as probabilistic reasoning to fuse individual results and verify their consistency it also attempts to learn novel actions performed by the user.

In [AL06b], a vision-based interface for controlling a computer mouse via 2D and 3D hand gestures is presented. Two vocabularies are defined: the first depends only on 2D hand tracking while the second makes use of 3D information and requires a second camera. The second condition of operation is of particular importance because it allows the gesture observer (a robot) to move along with the user. In another robotic application [KHB96], the user points the finger and extends the arm to indicate locations on the floor, in order to instruct a robot to move to the indicated location.

Applications where hand interaction facilitates the communication of a com-

mand or message from the user to the system, require that the posture and motion of hands is recognized and interpreted. Early gesture recognition applications supported just a few gestures that signified some basic commands or concepts to the computer system. For example in [DP93], a monocular vision system supported the recognition of a wide variance of yes/no hand gestures. In [SHJ94b], a rotation-invariant image representation was utilized to recognize a few hand gestures such as "hello" and "goodbye" in a controlled setup. The system in [CCK96], recognized simple natural gestures such a hand trajectories that comprised circles and lines.

Some systems combine the recognition of simple gestures with manipulative hand interaction. For example in [WO03], stereo-vision facilitates hand-tracking and gesture-recognition in a GUI that permits the user to perform window-management tasks, without the use of the mouse or keyboard. The system in [BPH98] integrated navigation control gestures into the "BattleView" virtual environment. The integrated gestures were utilized in navigating oneself as well as moving objects in the virtual environment. Hand-driven 3D manipulation and editing of virtual objects is employed in [PSH96, ZPD+97], in the context of a virtual environment for molecular biologists. In [SK98], a hand-gesture interface is proposed that allows the manipulation of objects in a virtual 3D environment by recognizing a few simple gestures and tracking hand motion. The system in [HCNP06] tracks hand motion to rotate the 3D content that is displayed in an autostereoscopic display. In the system of [Hoc98], the user interacts in front of a projection screen, and where interaction in physical space and pointing gestures are used to direct the scene for filmmaking.

In terms of communicative gestures, the sign language for the hearing impaired has received significant attention [SP95, CSW95, Wal95, GA97, SWP98, VM99, BH00, VM01, TSS02, YAT02, MWSK02]. Besides providing a constrained and meaningful dataset, it exhibits significant potential impact in society since it can facilitate the communication of the hearing impaired with machines through a natural modality for the user. In [ILI98], a bidirectional translation system between Japanese Sign Language and Japanese was implemented, in order to help the hearing impaired communicate with normal speaking people through sign language. Among the earliest systems is the one in [SP95] which recognizes about 40 American Sign Language which was later extended [SWP98] to observe the user's hands from a camera mounted on a cap worn by the user. Besides the recognition of individual hand postures, the system in [MWSK02] recognized motion primitives and full sentences, accounting for the fact that the same sign may have different meanings depending on context. The main difference of the system in [YAT02] is that it extracts motion trajectories from an image sequence and uses these trajectories as features in gesture recognition in combination with recognized hand postures.

Gestures have been utilized in the remote control of a television set via hand gestures in [Fre99], where an interface for video games is also considered. In [Koh97], a more general system for the control of home appliances was introduced. In [LK99], a gesture recognition method was developed to spot and recognize about 10 hand gestures for a human computer interface, instantiated to

control a slide presentation. The systems in [ZNG+04] and [CRMM00, MMR00], recognize a few hand postures for the control of in-car devices and non-safety systems, such as radio/CD, AC, telephone and navigation system, with hand postures and dynamic hand gestures, in an approach to simplify the interaction with these devices while driving. Relevant to control of electronic devices, in [MHP+01], a system is presenting for controlling a video camera via hand gestures with commands such as zoom, pan and tilt. In [TM96], a person-independent gesture interface was developed on a real robot; the user is able to issue commands such as how to grasp an object and where to put it. The application of gesture recognition in tele-operation systems has been investigated in [She93], to pinpoint the challenges that arise when controlling remote mechanisms in such large distances (earth to satellite) that the round trip time delay for visual feedback is several tenths of a second.

Tracking and recognizing body and hand motion has also been employed in personal training. The system in [BOP97] infers the posture of the whole body by observing the trajectories of hands and the head, in constrained setups. In [DB98], a prototype system for a virtual Personal Aerobics Trainer was implemented that recognizes stretching and aerobic movements and guides the user into a training program. Similarly in [Bec97], a virtual T'ai Chi trainer is presented. Recently, Sony [Fox05] introduced a system that tracks body motion against a uniform background and features a wide variety of gaming and personal training capabilities. The "ALIVE II" system [MDBP95] identifies full body gestures, in order to control "artificial life" creatures, such as virtual pets and companions that, sometimes, mimic the body gestures of the user. Gestures such as pointing the arm are interpreted by the simulated characters as a command to move to the indicated location. In addition, the user can issue gesture-driven commands to manipulate virtual objects. In [CT98] the authors present a hand and body gesture-driven interactive virtual environment for children.

The system in [Que00], attempts to recognize free-form hand gestures that accompany speech in natural conversations and which provide a complementary modality to speech for communication. A gesture-recognition application is presented in [JBMK97], where an automatic system for analyzing and annotating video sequences of technical presentations was developed. In this case, the system passively extracts information about the presenter of the talk. Gestures such as pointing or writing are recognized and utilized in the annotation of the video sequence. Similarly, in [BJ98a], a system that tracks the actions of the user on a blackboard actions was implemented. The system can recognize gestures that commands the system to e.g. "print", "save" and "cut" the contents of the blackboard.

3 The Proposed Approach to Human-Robot Interaction based on Hand Gestures

In this section we present the development of a prototype gesture recognition system intended for human-robot interaction. The application at hand involves natural interaction with autonomous robots installed in public places such as museums and exhibition centers. The operational requirements of such an application challenge existing approaches in that the visual perception system should operate efficiently under totally unconstrained conditions regarding occlusions, variable illumination, moving cameras, and varying background. Moreover, since no training of users can take place (users are assumed to be normal visitors of museums/exhibitions), the gesture vocabulary needs to be limited to a small number of natural, generic and intuitive gestures that humans use in their everyday human-to-human interactions.

The proposed gesture recognition system builds upon a probabilistic framework that allows the utilization of multiple information cues to efficiently detect regions belonging to human hands [BALT08]. The utilized information cues include color information, motion information through a background subtraction technique [GS99, SEG99], expected spatial location of hands within the image as well as velocity and shape of the detected hand segments. Tracking over time is achieved by a technique that can handle hands that may move in complex trajectories, occlude each other in the field of view of the robot's camera and vary in number over time [AL04b]. Finally, a simple set of hand gestures is defined based on the number of extended fingers and their spatial configuration.

3.1 The proposed approach in detail

A block diagram of the proposed gesture recognition system is illustrated in Figure 1. The first two processing layers of the diagram (i.e processing layers 1 and 2) perform the detection task (in the sense described in Section 2) while processing layers 3 and 4 correspond to the tracking and recognition tasks respectively. In the following sections, details on the implementation of the individual system components are provided.

3.1.1 Processing layer 1: Estimating the probability of observing a hand at the pixel level

Within the first layer, the input image is processed in order to identify pixels that depict human hands. Let $\mathcal{U}$ be the set of all pixels of an image. Let $\mathcal{M}$ be the subset of $\mathcal{U}$ corresponding to foreground pixels (i.e a human body) and $\mathcal{S}$ be the subset of $\mathcal{U}$ containing pixels that are skin colored. Accordingly, let $\mathcal{H}$ stand for the sets of pixels that depict human hands. The relations between the above mentioned sets are illustrated in the Venn diagram shown in Figure 2. The implicit assumption in the above formulation is that $\mathcal{H}$ is a subset of $\mathcal{M}$, i.e. hands always belong to the foreground. It is also important that according

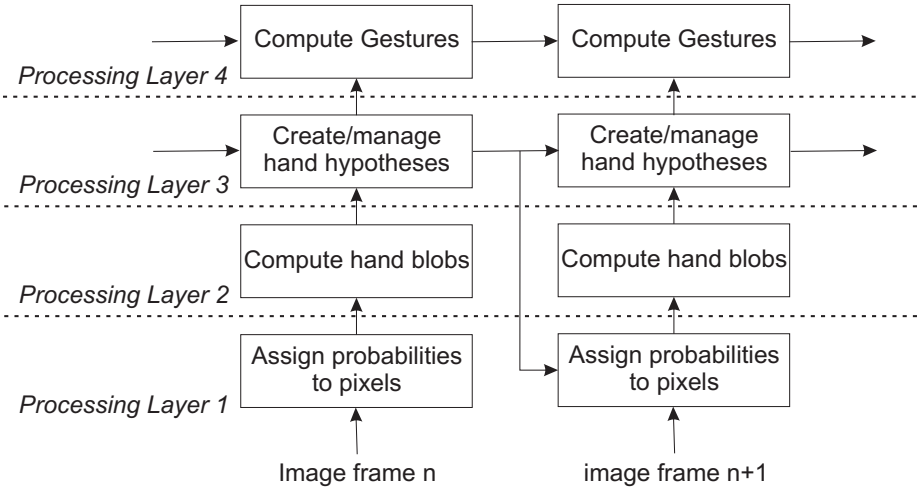


Fig. 1: Block diagram of the proposed approach for hand tracking and gesture recognition. Processing is organized into four layers.

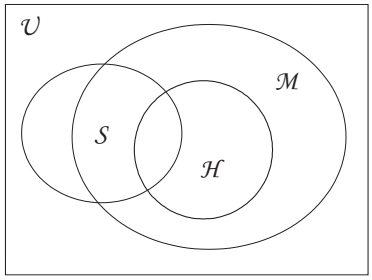


Fig. 2: The Venn diagram representing the relationship between the pixel sets $\mathcal{U}$, M, S and H.

to this model, all pixels belonging to hands are not necessarily assumed to be skin-colored.

Accordingly, let S, and H be binary random variables (i.e taking values in $\{0, 1\}$), indicating whether a pixel belongs to $\mathcal{S}$ and $\mathcal{H}$, respectively. Also, let M be a binary variable (determined by the employed foreground subtraction algorithm) that indicates whether a pixel belongs to $\mathcal{M}$. Let L be the 2D location vector containing the pixel image coordinates and let T be a variable that encodes a set of features regarding the currently tracked hypotheses (the contents of T will be explained later in this section). Given all the above, the goal of the this processing layer, is to compute whether a pixel belongs to a hand, given (a) the color c of a single pixel, (b) the information m on whether this pixel belongs to the background (i.e. $M = m$) and, (c) the values l and t of L and T, respectively. More specifically, the conditional probability

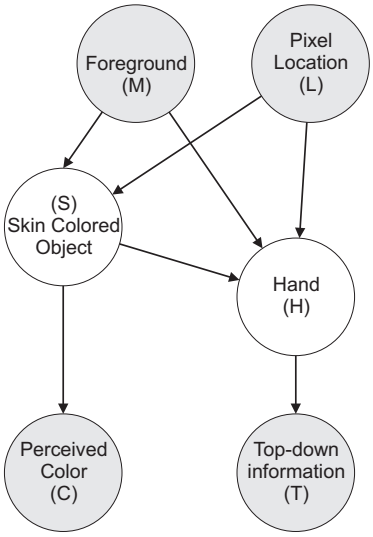


Fig. 3: The proposed Bayes net.

$P_h = P(H{=}1|C{=}c, T{=}t, L{=}l, M{=}m)$ needs to be estimated[1].

To perform this estimation, we assume the Bayesian network shown in Figure 3. The nodes in the graph of this figure correspond to random variables that represent degrees of belief on particular aspects of the problem. The edges in the graph are parameterized by conditional probability distributions that represent causal dependencies between the involved variables. It is known that

$$P(H{=}1|c,t,l,m) = \frac{P(H{=}1,c,t,l,m)}{P(c,t,l,m)} \tag{1}$$

By marginalizing the numerator over both possible values of S and the denominator over all four possible combinations of S and H (the values of S and H are expressed by the summation indices s and h, respectively), P_h can be expanded as:

$$P_h = \frac{\sum\limits_{s\in\{0,1\}} P(H{=}1,s,c,t,l,m)}{\sum\limits_{s\in\{0,1\}}\sum\limits_{h\in\{0,1\}} P(h,s,c,t,l,m)} \tag{2}$$

By applying the chain rule of probability and by taking advantage of the variable (in-)dependencies implied by the graph of Figure 3(b), we obtain:

$$P(h,s,c,t,l,m) = P(m)P(l)P(t|h)P(c|s)P(s|l,m)P(h|l,s,m) \tag{3}$$

[1] Note that capital letters are used to indicate variables and small letters to indicate specific values for these variables. For brevity, we will also use the notation $P(x)$ to refer to probability $P(X = x)$ where X any of the above defined variables and x a specific value of this variable.

Finally, by substituting to Equation (1), we obtain:

$$P_h = \frac{P(t|H=1) \sum_{s\in\{0,1\}} P(c|s)P(s|l,m)P(H{=}1|l,s,m)}{\sum_{h\in\{0,1\}} P(t|h) \sum_{s\in\{0,1\}} P(c|s)P(s|l,m)P(h|l,s,m)} \quad (4)$$

Details regarding the estimation of the individual probabilities that appear in Equation (4) are provided in the following sections.

Foreground segmentation
It can be easily verified that when $M = 0$ (i.e. a pixel belongs to the background), the numerator of Equation (4) becomes zero as well. This is because, as already mentioned, hands have been assumed to always belong to the foreground. This assumption simplifies computations because Equation (4) should only be evaluated for foreground pixels.

In order to compute M, we employ the foreground/background segmentation technique proposed by Stauffer and Grimson [GS99, SEG99] that employs an adaptive Gaussian mixture model on the background color of each image pixel. The number of Gaussians, their parameters and their weights in the mixture are computed online.

The color model
$P(c|s)$ is the probability of a pixel being perceived with color c given the information on whether it belongs to skin or not. To increase robustness against lighting variability, we transform colors to the YUV color space. Following the same approach as in [YLW98] and [AL04b], we completely eliminate the Y (luminance) component. This makes C a two-dimensional variable encoding the U and V (chrominance) components of the YUV color space.

$P(c|s)$ is obtained off-line through a separate training phase with the procedure described in [AL04b]. Assuming that C is discrete (i.e taking values in $[0..255]^2$) the result can be encoded in the form of two, 2D look-up tables; one table for skin-colored objects ($s = 1$) and one table for all other objects ($s = 0$). The rows and the columns of both look-up tables correspond to the U and V dimensions of the YUV color space.

The spatial distribution model
A spatial distribution model for skin and hands is needed in order to evaluate $P(s|l,m)$ and $P(h|l,s,m)$. These two probabilities express prior probabilities that can be obtained during training and are stored explicitly for each each location l (i.e for each image pixel). In order to estimate these probabilities, a set of four different quantities are computed off-line during training. These quantities are depicted in Table 1 and indicate the number of foreground pixels found in the training sequence for every possible combination of s and h. As discussed in Section 3.1.1, only computations for foreground pixels are necessary. Hence, all training data correspond to $M = 1$. We can easily express $P(s|l, M =$

Tab. 1: Quantities estimated during training for the spatial distribution model

h=0		h=1	
$s=0$	$s=1$	$s=0$	$s=1$
s_{00}	s_{01}	s_{10}	s_{11}

1) and $P(h|l,s,M=1)$ in terms of s_{00}, s_{01}, s_{10} and s_{11} as:

$$P(s|l,M{=}1|) = \frac{P(s,M{=}1,l)}{P(M{=}1,l)} = \frac{s_{0s}+s_{1s}}{s_{00}+s_{01}+s_{10}+s_{11}} \tag{5}$$

Similarly:

$$P(h|l,s,M{=}1|) = \frac{P(h,s,M{=}1,l)}{P(s,M{=}1,l)} = \frac{s_{hs}}{s_{0s}+s_{1s}} \tag{6}$$

Top-down information regarding hand features
Within the second and the third processing layers, pixel probabilities are converted to blobs (second layer) and hand hypotheses which are tracked over time (third layer). These processes are described later in Sections 3.1.2 and 3.1.3, respectively. Nevertheless, as Figure 1 shows, the third processing layer of image n provides top-down information exploited during the processing of image $n+1$ at layer 1. For this reason, the description of the methods employed to compute the probabilities $P(t|h)$ that are further required to estimate P_h is deferred to section 3.1.3.

3.1.2 Processing layer 2: From pixels to blobs

This layer applies hysteresis thresholding on the probabilities determined at layer 1. These probabilities are initially thresholded by a "strong" threshold T_{max} to select all pixels with $P_h > T_{max}$. This yields high-confidence hand pixels that constitute the seeds of potential hand blobs. A second thresholding step, this time with a "weak" threshold T_{min}, along with prior knowledge with respect to object connectivity to form the final hand blobs. During this step, pixels with probability $P_h > T_{min}$ where $T_{min} < T_{max}$, that are immediate neighbors of hand pixels are recursively added to each blob.

A connected components labeling algorithm is then used to assign different labels to pixels that belong to different blobs. Size filtering on the derived connected components is also performed to eliminate small, isolated blobs that are attributed to noise and do not correspond to meaningful hand regions.

Finally, a feature vector for each blob is computed. This feature vector contains statistical properties regarding the spatial distribution of pixels within the blob and will be used within the next processing layer for data association.

3.1.3 Processing layer 3: From blobs to object hypotheses

Within the third processing layer, blobs are assigned to hand hypotheses which are tracked over time. Tracking over time is realized through a scheme which

can handle multiple objects that may move in complex trajectories, occlude each other in the field of view of a possibly moving camera and whose number may vary over time. For the purposes of this paper[2], it suffices to mention that a hand hypothesis h_i is essentially represented as an ellipse $h_i = h_i\,(c_{x_i}, c_{y_i}, \alpha_i, \beta_i, \theta_i)$ where (c_{x_i}, c_{y_i}) is the ellipse centroid, α_i and β_i are, respectively, the lengths of the major and minor axis of the ellipse, and θ_i is its orientation on the image plane. The parameters of each ellipse are determined by the covariance matrix of the locations of blob pixels that are assigned to a certain hypothesis. The assignment of blob pixels to hypotheses ensures (a) the generation of new hypotheses in cases of unmatched evidence (unmatched blobs), (b) the propagation and tracking of existing hypotheses in the presence of multiple, potential occluding objects and (c) the elimination of invalid hypotheses (i.e. when tracked objects disappear from the scene of view).

Top-down information regarding hand features revisited
In this work, for each tracked hand hypothesis, a feature vector T is generated which is propagated in a "top-down" direction in order to further assist the assignment of hand probabilities to pixels at processing layer 1. The feature vector T consists of two different features:

1. The average vertical speed v of a hand, computed as the vertical speed of the centroid of the ellipse modeling the hand. The rationale behind the selection of this feature is that hands are expected to exhibit considerable average speed v compared to other skin colored regions such as heads.

2. The ratio r of the perimeter of the hand contour over the circumference of a hypothetical circle having the same area as the area of the hand. The rationale behind the selection of this feature is that hands are expected to exhibit high r compared to other objects. That is, $r = \frac{1}{2}\rho/\sqrt{\pi\alpha}$, where ρ and α are the hand circumference and area, respectively.

Given v and r, $P(t|h)$ is approximated as:

$$P(t|h) \approx P(v|h)P(r|h) \tag{7}$$

$P(t|h)$ is the probability of measuring a specific value t for the feature vector T, given the information of whether a pixel belongs to a hand or not. A pixel is said to belong to a hand, depending on whether its image location lies within the ellipse modeling the hand hypothesis. That is, the feature vector T encodes a set of features related to existing (tracked) hands that overlap with the pixel under consideration.

In our implementation, both $P(v|h)$ and $P(r|h)$ are given by means of one-dimensional look-up tables that are computed off-line, during training. If there is more than one hypothesis overlapping with the specific pixel under consideration, the hypothesis that yields maximal results is chosen for $P(t|h)$. Moreover, if there is no overlapping hypothesis at all, all of the conditional probabilities

[2] For the details of this tracking process, the interested reader is referred to [AL04b]

(a) (b) (c) (d)

Fig. 4: The gesture vocabulary of the proposed approach. (a) The "Stop" gesture, (b) The "Thumbs Up" gesture. (c) The "Thumbs Down" gesture. (d) The "Point" gesture.

of Equation (7) are substituted by the maximum values of their corresponding look-up tables.

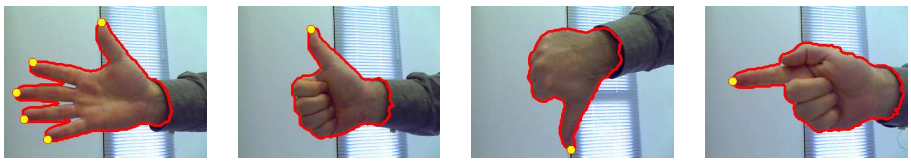
3.1.4 Processing layer 4: Recognizing hand gestures

The application considered in this paper involves natural interaction with autonomous mobile robots installed in public places such as museums and exhibition centers. Since the actual users of the system will be untrained visitors of a museum/exhibition, gestures should be as intuitive and natural as possible. Moreover, the challenging operational requirements of the application at hand impose the absolute need for gestures to be simple and robustly interpretable. Four simple gestures have been chosen to comprise the proposed gesture vocabulary which is graphically illustrated in Figure 4. All four employed gestures are static gestures, i.e., gestures in which the information to be communicated lies in the hand and finger posture at a certain moment in time. More specifically, the employed gestures are:

- The "Stop" gesture. The user extends his/her hand with all five fingers stretched to stop the robot from its current action.
- The "Thumbs Up" gesture. The user performs a "thumbs up" sign to approve or answer "yes" to a question by the robot.
- The "Thumbs Down" gesture. The user expresses disapproval or answers "no" to a question by doing the thumbs down gesture.
- The "Point" gesture. The user points to a specific exhibit or point of interest to ask the robot to guide him/her there.

It is also important that because of the generic nature of the employed gestures, their actual meaning can be interpreted by the robot based on specific, contextual information related to the scenario of use.

In order to robustly recognize the gestures constituting our gesture vocabulary, we employ a rule-based technique that relies on the number and the posture of the distinguishable fingers i.e the number of detected fingertips corresponding



(a) (b) (c) (d)

Fig. 5: Fingertip Detection. Fingers are denoted as black yellow circles

to each tracked hand hypothesis and their relative location with respect to the centroid of the hypothesis.

Finger Detection

Fingertip detection is performed by evaluating a curvature measure of the contour of the blobs that correspond to each hand hypothesis as in [AL06b]. The employed curvature measure assumes values in the range $[0.0, 1.0]$ and is defined as:

$$K_l(P) = \frac{1}{2}\left(1 + \frac{\overrightarrow{P_1P} \cdot \overrightarrow{P_2P}}{\|\overrightarrow{P_1P}\| \cdot \|\overrightarrow{P_2P}\|}\right) \tag{8}$$

where P_1, P and P_2 are successive points on the contour, P being separated from P_1 and P_2 by the same number of contour points. The symbol $(\cdot)$ denotes the vector dot product. The algorithm for finger detection computes $K_l(P)$ for all contour points of a hand and at various scales (i.e. for various values of the parameter l). A contour point P is then characterized as the location of a fingertip if both of the following conditions are met:

- $K_l(P)$ exceeds a certain threshold for at least one of the examined scales, and,
- $K_l(P)$ is a local maximum in its (scale-dependent) neighborhood of the contour.

Evaluation of curvature information on blob contours points has been demonstrated in the past[AL06b] to be a robust way to detect fingertips.

A significant advantage of contour features like fingertips is that in most cases they can be robustly extracted regardless the size of the blob (i.e distance of the observer), lighting conditions and other parameters that usually affect color and appearance based features. Figure 5 shows some examples from a fingertip detection experiment. In this experiment, there exist several hands which are successfully tracked among images. Fingers are also detected and marked with black squares. In the reported experiments, the curvature threshold of the first criterion was set to 0.7.

Recognizing a Gesture

As already mentioned, all employed gestures are static i.e., gestures in which the information to be communicated lies in features obtained at a specific moment

Tab. 2: Rules used to recognize the four gestures of our vocabulary.

Gesture	Visible Fingertips	Orientation ϕ (in degrees)
Stop	5	Irrelevant
Thumbs Up	1	$\phi \in [60, 120]$
Thumbs Down	1	$\phi \in [240, 300]$
Point	1	$\phi \in [0, 60] \cup [120, 240] \cup [300, 360]$

in time. The employed features consist of the number of distinguishable fingers (i.e fingers with distinguishable fingertips) and their orientation ϕ with respect to the horizontal image axis. To compute the orientation ϕ of a particular finger, the vector determined by the hand's centroid and the corresponding fingertip is assumed.

To recognize the four employed gestures a rule based approach is used. Table 2 summarizes the rules that need to be met for each of the four gestures in our vocabulary. Moreover, to determine the specific point in time that a gesture takes place three additional criteria have to be satisfied.

- Criterion 1: The hand posture has to last for at least a fixed amount of time t_g. In the actual implementation of the system, a minimum duration of half a second is employed (i.e $t_g = 0.5$ sec). Assuming a frame rate of 30Hz, this means that in order to recognize a certain posture, this has to be maintained for a minimum of fifteen consecutive image frames.

- Criterion 2: The hand that performs the gesture has to be (almost) still. This is determined by applying the requirement that the hand centroid remains within a specific threshold radius r_g for at least t_g seconds. In all our experiments an r_g value of about 30 pixels has been proven sufficient to ensure that the hand is almost standstill.

- Criterion 3. The speed of the hand has to be at its minimum with respect to time. To determine whether the hand speed has reached its minimum, a time lag t_l is assumed (fixed to about 0.3 sec in our experiments).

3.2 Experimental results

The proposed approach has been assessed using several video sequences containing people performing various gestures in indoor environments. Several videos of example runs are available on the web[3].

In this section we will present results obtained from a sequence depicting a man performing a variety of hand gestures in a setup that is typical for human robot interaction applications. i.e the subject is standing at a typical distance of about 1m from the robot looking towards the robot. The robot's camera is installed at a distance of approximately 1.2m from the floor. The resolution of the sequence is 640×480 and it was obtained with a standard, low-end web camera at 30 frames per second. Figure 6 depicts various intermediate

[3] http://www.ics.forth.gr/ xmpalt/research/gestures/index.html

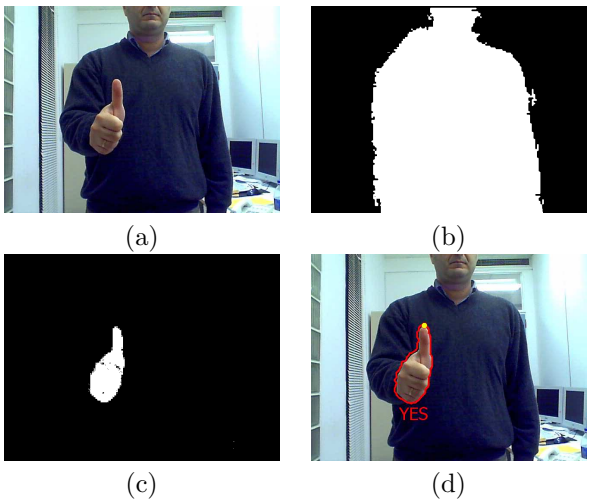


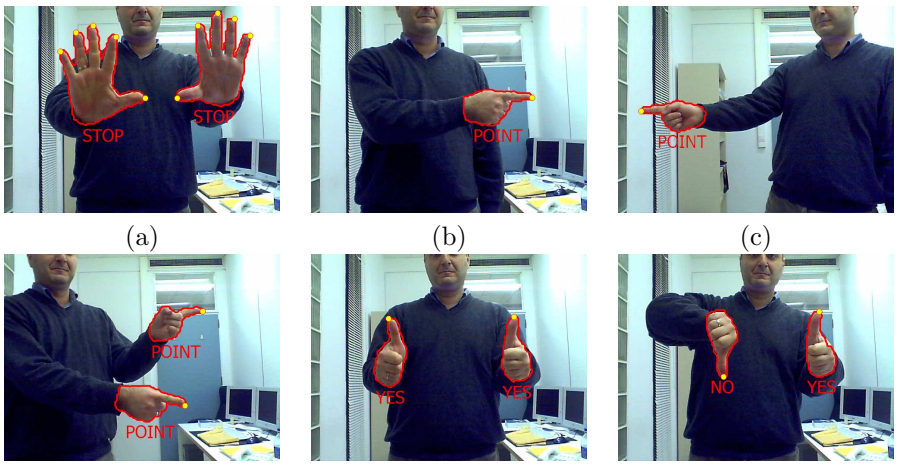
Fig. 6: The proposed approach in operation. (a) original frame, (b) background subtraction result, (c), pixel probabilities for hands, (d),contour and fingertip detection

results obtained at different stages of the proposed approach. A frame of the test sequence is shown in Figure 6(a). Figure 6(b) depicts the result of the background subtraction algorithm, i.e $P(M)$. In order to achieve real-time performance, the background subtraction algorithm operates at down-sampled images of dimensions 160×120. Figure 6(c) depicts P_h i.e. the result of the first processing layer of the proposed approach. The contour of the blob and the detected fingertip that correspond to the only present hand hypothesis is shown in Figure 6(d). As can be verified, the algorithm manages to correctly identify the hand of the depicted man. Notice also that, in contrast to what would happen if only color information were utilized, neither skin-colored objects in the background nor the subject's face is falsely recognized as a hand.

Figure 7 shows six more frames out of the same sequence. In all cases, the proposed approach has been successful in correctly identifying the hands of the person and in correctly recognizing the performed gesture. The presented results were obtained at a standard 3GHz personal computer which was able to process images of size 640×480 at 30Hz.

4 Summary

In this paper, we reviewed several existing methods for supporting vision-based human-computer interaction based on the recognition of hand gestures. The provided review covers research work related to all three individual subproblems of the full problem, namely detection, tracking and recognition. Moreover, we provide an overview of some integrated gesture recognition systems.



(a) (b) (c)

(d) (e) (f)

Fig. 7: Six frames of a sequence depicting a man performing gestures in an office environment.

Additionally, in this paper we have presented a novel gesture recognition system intended for natural interaction with autonomous robots that guide visitors in museums and exhibition centers. The proposed gesture recognition system builds on a probabilistic framework that allows the utilization of multiple information cues to efficiently detect image regions that belong to human hands. Tracking over time is achieved by a technique that can simultaneously handle multiple hands that may move in complex trajectories, occlude each other in the field of view of the robot's camera and vary in number over time. Dependable hand tracking, combined with fingertip detection, facilitates the definition of a small and simple hand gesture vocabulary that is both robustly interpretable and intuitive to humans interacting with robots. Experimental results presented in this paper, confirm the effectiveness and the efficiency of the proposed approach, meeting the run-time requirements of the task at hand. Nevertheless, and despite the vast amount of relevant research efforts, the problem of efficient and robust vision-based recognition of natural hand gestures in unprepared environments still remains open and challenging, and is expected to remain of central importance to the computer vision community in the forthcoming years.

Acknowledgements

This work has been partially supported by EU-IST NoE MUSCLE (FP6-507752), the Greek national GSRT project XENIOS and the EU-IST project INDIGO (FP6-045388).

References

[ADS98] Y. Azoz, L. Devi, and R. Sharma. Reliable tracking of human arm dynamics by multiple cue integration and constraint fusion. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 905–910, Santa Barbara, CA, 1998.

[Ahm94] S. Ahmad. A usable real-time 3D hand tracker. In *Asilomar Conference on Signals, Systems and Computers*, pages 1257–1261, Pacific Grove, CA, 1994.

[AL04a] A. A. Argyros and M. I. A. Lourakis. 3D tracking of skin-colored regions by a moving stereoscopic observer. *Applied Optics*, 43(2):366–378, January 2004.

[AL04b] A. A. Argyros and M. I. A. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In *Proc. European Conference on Computer Vision*, pages 368–379, Prague, Chech Republic, May 2004.

[AL06a] A. A. Argyros and M. I. A. Lourakis. Binocular hand tracking and reconstruction based on 2D shape matching. In *Proc. International Conference on Pattern Recognition (ICPR)*, Hong-Kong, China, 2006.

[AL06b] A. A. Argyros and M. I. A. Lourakis. Vision-based interpretation of hand gestures for remote control of a computer mouse. In *ECCV Workshop on HCI*, pages 40–51, Graz, Austria, May 2006.

[AP96] A. Azarbayejani and A. Pentland. Real-time self-calibrating stereo person-tracker using 3-d shape estimation from blob features. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 99–108, Vienna, Switzerland, 1996.

[AS01] V. Athitsos and S. Sclaroff. 3D hand pose estimation by finding appearance-based matches in a large database of training views. In *IEEE Workshop on Cues in Communication*, pages 100–106, 2001.

[AS02] V. Athitsos and S. Sclaroff. An appearance-based framework for 3D hand shape classification and camera viewpoint estimation. In *IEEE Conference on Face and Gesture Recognition*, pages 45–50, Washington, DC, 2002.

[AS03] V. Athitsos and S. Sclaroff. Estimating 3D hand pose from a cluttered image. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 432–439, Madison, WI, 2003.

[BALT08] H. Baltzakis, A. Argyros, M. Lourakis, and P. Trahanias. Tracking of human hands and faces through probabilistic fusion of multiple visual cues. In *Proc. International Conference on Computer Vision Systems (ICVS), to appear*, Santorini, Greece, May 2008.

[BBC93] M. Brand, L. Birnbaum, and P. Cooper. Sensible scenes: Visual understanding of complex structures through causal analysis. In *AAAI Conference*, pages 45–56, 1993.

[BD96] A. Bobick and J. Davis. Real-time recognition of activity using temporal templates. In *IEEE Workshop on Applications of Computer Vision*, pages 39–42, Sarasota, FL, 1996.

[BD00] G. Bradski and J. Davis. Motion segmentation and pose recognition with motion history gradients. In *IEEE Workshop on Applications of Computer Vision*, pages 238–244, Palm Springs, CA, 2000.

[BD01] A. Bobick and J. Davis. The representation and recognition of action using temporal templates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 3(3):257–267, 2001.

[BD02] G. Bradski and J. Davis. Motion segmentation and pose recognition with motion history gradients. *Machine Vision and Applications*, 13(3):174–184, 2002.

[Bec97] D. A. Becker. Sensei: A real-time recognition, feedback, and training system for T'ai chi gestures. 1997.

[BF95] U. Brockl-Fox. Real-time 3D interaction with up to 16 degrees of freedom from monocular image flows. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 172–178, Zurich, Switzerland, 1995.

[BH94] A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *Proc. European Conference on Computer Vision*, volume 1, pages 299–308, Stocklholm, Sweden, 1994.

[BH00] B. Bauer and H. Hienz. Relevant features for video-based continuous sign language recognition. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 440–445, 2000.

[BHW+05] C. Bauckhage, M. Hanheide, S. Wrede, T. Kaster, M. Pfeiffer, and G. Sagerer. Vision systems with the human in the loop. *EURASIP Journal on Applied Signal Processing*, 14:2375–2390, 2005.

[BHWS04] C. Bauckhage, M. Hanheide, S. Wrede, and G. Sagerer. A cognitive vision system for action recognition in office environments. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 827–833, 2004.

[BJ96] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. In *Proc. European Conference on Computer Vision*, pages 329–342, 1996.

[BJ98a] M. Black and A. Jepson. A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gesture and expression. In *Proc. European Conference on Computer Vision*, volume 2, pages 909–924, 1998.

[BJ98b] M. Black and A. Jepson. Recognizing temporal trajectories using the condensation algorithm. In *IEEE Int. Conference on Automatic Face and Gesture Recognition*, pages 16–21, 1998.

[BJ98c] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision*, 26(1):63–84, 1998.

[BK98] M. Breig and M. Kohler. Motion detection and tracking under constraint of pan-tilt cameras for vision-based human computer interaction. Technical Report 689, Informatik VII, University of Dortmund/Germany, August 1998.

[BM92] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[BMM97] H. Birk, T. B. Moeslund, and C. B. Madsen. Real-time recognition of hand alphabet gestures using principal component analysis. In *Proc. Scandinavian Conference on Image Analysis*, Lappeenranta, Finland, June 1997.

[BMP02] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[BNI99] A. Blake, B. North, and M. Isard. Learning multi-class dynamics. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, volume 11, pages 389–395, 1999.

[BOP97] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 994–999, San Juan, Puerto Rico, June 1997.

[Bor88] G. Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 10(6):849–865, 1988.

[BPH98] G. Berry, V. Pavlovic, and T. Huang. Battleview: A multimodal hci research application. In *Workshop on Perceptual User Interfaces*, pages 67–70, San Francisco, CA, 1998.

[Bra98] G. Bradski. Real time face and object tracking as a component of a perceptual user interface. In *IEEE Workshop on Applications of Computer Vision*, pages 214–219, 1998.

[Bre97] C. Bregler. Learning and recognizing human dynamics in video sequences. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 568–574, Puerto Rico, 1997.

[BTW77] H. Barrow, R. Tenenbaum, J. Bolles, and H. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Int. Joint Conference in Artificial Intelligence*, pages 659–663, 1977.

[BW97] A. F. Bobick and A. D. Wilson. A state-based approach to the representation and recognition of gesture. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(12):1325–1337, 1997.

[CBA+96a] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick, and A. Pentland. Invariant features for 3-d gesture recognition. In *IEEE Int. Conf. Automatic Face and Gesture Recognition*, pages 157–162, Killington, VT, 1996.

[CBA+96b] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick, and A. Pentland. Invariant features for 3-d gesture recognition. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 157–162, Killington, Vermont, USA, October 1996.

[CBC95] J. Crowley, F. Berard, and J. Coutaz. Finger tracking as an input device for augmented reality. In *International Workshop on Gesture and Face Recognition*, Zurich, June 1995.

[CCK96] C. Cohen, L. Conway, and Dan Koditschek. Dynamical system representation, generation, and recognition of basic oscillatory motion gestures. In *International Conference on Automatic Face and Gesture Recognition*, Killington, VT, 1996.

[CG99] J. Cai and A. Goshtasby. Detecting human faces in color images. *Image and Vision Computing*, 18(1):63–75, 1999.

[CH96] A. Colmenarez and T. Huang. Maximum likelihood face detection. In *Int. Conference on Automatic Face and Gesture Recognition*, pages 307–311, Killington, VT, 1996.

[Che95] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(8):790–799, 1995.

[CJ92] T. F. Cootes and Taylor C. J. Active shape models - smart snakes. In *British Machine Vision Conference*, pages 266–275, 1992.

[CJHG95] T. F. Cootes, Taylor C. J., Cooper D. H., and J. Graham. Active shape models - their training and applications. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[CKBH00] G. Cheung, T. Kanade, J. Bouguet, and M. Holler. A real time system for robust 3D voxel reconstruction of human motions. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 714–720, 2000.

[CL01] H. Chen and T. Liu. Trust-region methods for real-time tracking. In *Proc. International Conference on Computer Vision (ICCV)*, volume 2, pages 717–722, Vancouver, Canada, 2001.

[CN98] D. Chai and K. Ngan. Locating the facial region of a head and-shoulders color image. In *IEEE Int. Conference on Automatic Face and Gesture Recognition*, pages 124–129, Piscataway, NJ, 1998.

[CPC06] M. Cote, P. Payeur, and G. Comeau. Comparative study of adaptive segmentation techniques for gesture analysis in unconstrained environments. In *IEEE Int. Workshop on Imagining Systems and Techniques*, pages 28–33, 2006.

[CRM00] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 142–149, Hilton Head Island, SC, 2000.

[CRM03] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(5):564– 577, 2003.

[CRMM00] N. Cairnie, I. Ricketts, S. McKenna, and G. McAllister. Using finger-pointing to operate secondary controls in an automobile. In *Intelligent Vehicles Symposium*, volume 4, pages 550–555, Dearborn, MI, 2000.

[CSW95] Y. Cui, D. Swets, and J. Weng. Learning-based hand sign recognition using shoslf-m. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 201–206, Zurich, 1995.

[CT98] R. Cutler and M. Turk. View-based interpretation of real-time optical flow for gesture recognition. In *Proc. International Conference on Face and Gesture Recognition*, pages 416–421, Washington, DC, USA, 1998. IEEE Computer Society.

[CW96a] Y. Cui and J. Weng. Hand segmentation using learning-based prediction and verification for hand sign recognition. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 88–93, 1996.

[CW96b] Y. Cui and J. Weng. Hand sign recognition from intensity image sequences with complex background. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 88–93, 1996.

[Dav01] J. Davis. Hierarchical motion history images for recognizing human motion. In *IEEE Workshop on Detection and Recognition of Events in Video*, pages 39–46, Vancouver, Canada, 2001.

[DB98] J. Davis and A. Bobick. Virtual pat: A virtual personal aerobic trainer. In *Workshop on Perceptual User Interfaces*, pages 13–18, 1998.

[DBR00] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 126–133, Hilton Head Island, SC, 2000.

[DD91] A. Downton and H. Drouet. Image analysis for model-based sign language coding. In *Int. Conf. Image Analysis and Processing*, pages 637–644, 1991.

[DEP96] T. Darrell, I. Essa, and A. Pentland. Task-specific gesture analysis in real-time using interpolated views. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(12):1236–1242, 1996.

[DKS01] S. M. Dominguez, T. Keaton, and A. H. Sayed. A robust finger tracking method for wearable computer interfacing. *IEEE Transactions on Multimedia*, 8(5):956–972, 2001.

[DP93] T. Darrell and A. Pentland. Space-time gestures. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 335–340, New York, NY, 1993.

[DP95] T. Darrell and A. Pentland. Attention driven expression and gesture analysis in an interactive environment. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 135–140, Zurich, Switzerland, 1995.

[DS92] Terzopoulos D. and R. Szeliski. *Tracking with Kalman Snakes*, pages 3–20. MIT Press, 1992.

[DS94a] J. Davis and M. Shah. Recognizing hand gestures. In *Proc. European Conference on Computer Vision*, pages 331–340, 1994.

[DS94b] J. Davis and M. Shah. Visual gesture recognition. *Vision, Image, and Signal Processing*, 141(2):101–106, 1994.

[DWT04] K. Derpanis, R. Wildes, and J. Tsotsos. *Hand Gesture Recognition within a Linguistics-Based Framework*, volume 3021 of *LCNS*, pages 282–296. Springer Berlin / Heidelberg, 2004.

[EGG+03] J. Eisenstein, S. Ghandeharizadeh, L. Golubchik, C. Shahabi, Donghui Y., and R. Zimmermann. Device independence and extensibility in gesture recognition. In *IEEE Virtual Reality*, pages 207–214, 2003.

[EKR+98] S. Eickeler, A. Kosmala, G. Rigoll, A. Jain, S. Venkatesh, and B. Lovell. Hidden markov model based continuous online gesture recognition. In *International Conference on Pattern Recognition*, volume 2, pages 1206–1208, 1998.

[ETK91] M. Etoh, A. Tomono, and F. Kishino. Stereo-based description by generalized cylinder complexes from occluding contours. *Systens and Computers in Japan*, 22(12):79–89, 1991.

[FAK03] H. Fillbrandt, S. Akyol, and K. F. Kraiss. Extraction of 3D hand shape and posture from images sequences from sign language recognition. In *Proc. International Workshop on Analysis and Modeling of Faces and Gestures*, pages 181–186, Nice, France, October 2003.

[FB02] R. Fablet and M. Black. Automatic detection and tracking of human motion with a view-based representation. In *Proc. European Conference on Computer Vision*, pages 476–491, Berlin, Germany, 2002.

[FHR00] J. Friedman, T. Hastie, and Tibshiranim R. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28(2):337–374, 2000.

[FM99] R. Francois and G. Medioni. Adaptive color background modeling for real-time segmentation of video streams. In *Int. Conferebce on Imaging Science, Systems, and Technology*, pages 227–232, Las Vegas, NA, 1999.

[Fox05] B. Fox. Invention: Magic wand for gamers. *New Scientist*, August 2005.

[FR95] W. Freeman and M. Roth. Orientation histograms for hand gesture recognition. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 296–301, Zurich, Switzerland, 1995.

[Fre99] W. Freeman. Computer vision for television and games. In *Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems.*, page 118, 1999.

[FS97] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[FSM94] M. Fukumoto, Y. Suenaga, and K. Mase. "finger-pointer": Pointing interface by image processing. *Computers and Graphics*, 18(5):633–642, 1994.

[FW95] W. Freeman and C. Weissman. Television control by hand gestures. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 179–183, Zurich, Switzerland, 1995.

[GA97] K. Grobel and M. Assan. Isolated sign language recognition using hidden markov models. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 162–167, 1997.

[GD95] D. Gavrila and L. Davis. Towards 3D model-based tracking and recognition of human movement: A multi-view approach. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 272–277, Zurich, Switzerland, 1995.

[GD96] D. Gavrila and L. Davis. 3-D model-based tracking of humans in action: a multi-view approach. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 73–80, 1996, 1996.

[GdBUP95] L. Goncalves, E. di Bernardo, E. Ursella, and P. Perona. Monocular tracking of the human arm in 3D. In *Proc. International Conference on Computer Vision (ICCV)*, pages 764–770, Cambridge, 1995.

[GMR+02] N. Gupta, P. Mittal, S. Roy, S. Chaudhury, and S. Banerjee. Condensation-based predictive eigentracking. In *Indian Conference on Computer Vision, Graphics and Image Processing*, Ahmadabad,India, December 2002.

[GS99] W. E. L. Grimson and C. Stauffer. Adaptive background mixture models for real time tracking. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 246–252, Ft. Collins, USA, June 1999.

[GWP99] S. Gong, M. Walter, and A. Psarrou. Recognition of temporal structures: Learning prior and propagating observation augmented densities via hidden markov states. In *Proc. International Conference on Computer Vision (ICCV)*, pages 157–162, 1999.

[HB96] G. Hager and P. Belhumeur. Real-time tracking of image regions with changes in geometry and illumination. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 403–410, Washington, DC, 1996.

[HCNP06] K. Hopf, P. Chojecki, F. Neumann, and D. Przewozny. Novel autostereoscopic single-user displays with user interaction. In *SPIE*, volume 6392, Boston, MA, 2006.

[HH96a] A. Heap and D. Hogg. 3D deformable hand models. In *Gesture Workshop on Progress in Gestural Interaction*, pages 131–139. Springer-Verlag, 1996.

[HH96b] T. Heap and D. Hogg. Towards 3D hand tracking using a deformable model. In *IEEE Int. Conf. Automatic Face and Gesture Recognition*, pages 140–145, Killington, VT, 1996.

[HKR93] D. Huttenlocher, G. Klanderman, and W. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(9):850–863, 1993.

[HLCP02] C. Hue, J. Le Cadre, and P. Perez. Sequential monte carlo methods for multiple target tracking and data fusion. *IEEE Transactions on Signal Processing*, 50:309–325, 2002.

[Hoc98] M. Hoch. A prototype system for intuitive film planning. In *Automatic Face and Gesture Recognition*, pages 504–509, Nara, Japan, 1998.

[HS95] A. Heap and F. Samaria. Real-time hand tracking and gesture recognition using smart snakes. In *Interface to Real and Virtual Worlds*, Montpellier, 1995.

[HVD+99a] R. Herpers, G. Verghese, K. Darcourt, K. Derpanis, R. Enenkel, J. Kaufman, M. Jenkin, E. Milios, A. Jepson, and J. Tsotsos. An active stereo vision system for recognition of faces and related hand gestures. In *Int. Conf. on Audio- and Video-based Biometric Person Authentication*, pages 217–223, Washington, D. C., 1999.

[HVD+99b] R. Herpers, G. Verghese, K. Derpanis, R. McCready, J. MacLean, A. Levin, D. Topalovic, L. Wood, A. Jepson, and

J. Tsotsos. In *International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 96–104, Corfu, Greece, 1999.

[IB96a] M. Isard and A. Blake. Contour tracking by sotchastic propagation of conditional density. In *Proc. European Conference on Computer Vision*, pages 343–356, Cambridge, UK, 1996.

[IB96b] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conference on Computer Vision*, pages 343–356, Cambridge, UK, April 1996.

[IB98a] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *Int. Journal of Computer Vision*, 29(1):5–28, 1998.

[IB98b] M. Isard and A. Blake. Icondensation: unifying low-level and high-level tracking in a stochastic framework. In *Proc. European Conference on Computer Vision*, pages 893–908, Berlin, Germany, 1998.

[IB98c] M. Isard and A. Blake. A mixed-state condensation tracker with automatic model-switching. In *Proc. International Conference on Computer Vision (ICCV)*, pages 107–112, 1998.

[ICLB05] B. Ionescu, D. Coquin, P. Lambert, and V. Buzuloiu1. Dynamic hand gesture recognition using the skeleton of the hand. *EURASIP Journal on Applied Signal Processing*, 13:2101–2109, 2005.

[ILI98] K. Imagawa, S. Lu, and S. Igi. Color-based hands tracking system for sign language recognition. In *Int. Conf. Face and Gesture Recognition,*, pages 462–467, 1998.

[IM01] M. Isard and J. MacCormick. Bramble: a bayesian multipleblob tracker. In *Proc. International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, 2001.

[JBMK97] S. Ju, M. Black, S. Minneman, and D. Kimber. Analysis of gesture and action in technical talks for video indexing. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 595–601, 1997.

[Jen99] C. Jennings. Robust finger tracking with multiple cameras. In *IEEE workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, pages 152–160, Corfu, Greece, 1999.

[JP97] T. Jebara and A. Pentland. Parametrized structure from motion for 3D adaptive feedback tracking of faces. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 144–150, Piscataway, NJ, 1997.

[JR02] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.

[JRP97] T. Jebara, K. Russel, and A. Pentland. Mixture of eigenfeatures for real-time structure from texture. In *Proc. International Conference on Computer Vision (ICCV)*, pages 128–135, Piscataway, NJ, 1997.

[Kal60] R. E. Kalman. A new approach to linear flitering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82:35–42, 1960.

[Kam98] M. Kampmann. Segmentation of a head into face, ears, neck and hair forknowledge-based analysis-synthesis coding of videophone sequences. In *Proc. International Conference on Image Processing (ICIP)*, volume 2, pages 876–880, Chicago, IL, 1998.

[KF94] W. Krueger and B. Froehlich. The responsive workbench. *IEEE Computer Graphics and Applications*, 14(3):12–15, 1994.

[KH95] J. Kuch and T. Huang. Vision based hand modeling and tracking for virtual teleconferencing and telecollaboration. In *Proc. International Conference on Computer Vision (ICCV)*, pages 666–671, 1995.

[KHB96] D. Kortenkamp, E. Huber, and R. Bonasso. Recognizing and interpreting gestures on a mobile robot. In *National Conference on Artificial Intelligence*, 1996, 1996.

[KI91] S. Kang and K. Ikeuchi. A framework for recognizing grasps. Technical Report CMU-RI-TR-91-24, Robotics Institute, Carnegie Mellon University, November 1991.

[KI93] S. Kang and K. Ikeuchi. Toward automatic robot instruction for perception - recognizing a grasp from observation. *IEEE Transactions on Robotics and Automation*, 9:432–443, 1993.

[KK96] R. Kjeldsen and J. Kender. Finding skin in color images. In *IEEE Int. Conf. Automatic Face and Gesture Recognition*, pages 312–317, Killington, VT, 1996.

[KKAK98] S. Kim, N. Kim, S. Ahn, and H. Kim. Object oriented face detection using range and color information. In *IEEE Int. Conference on Automatic Face and Gesture Recognition*, pages 76–81, Piscataway, NJ, 1998.

[KL01] W. Kim and J. Lee. Visual tracking using snake for object's discrete motion. In *IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2608–2613, Seoul, Korea,, 2001.

[KM03] H. Kawashima and T. Matsuyama. Multi-viewpoint gesture recognition by an integrated continuous state machine. *Systems and Computers in Japan*, 34(14):1–12, 2003.

[KMA01] E. Koller-Meier and F. Ade. Tracking multiple objects using the condensation algorithm. *Journal of Robotics and Autonomous Systems*, 34(3):93–105, 2001.

[KMB94] I. Kakadiaris, D. Metaxas, and R. Bajcsy. Active partdecomposition, shape and motion estimation of articulated objects: A physics-based approach. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 980–984, 1994.

[Koh97] M. Kohler. System architecture and techniques for gesture recognition inunconstrained environments. In *International Conference on Virtual Systems and MultiMedia*, volume 10-12, pages 137–146, 1997.

[KOKS01] T. Kurata, T. Okuma, M. Kourogi, and K. Sakaue. The hand mouse: Gmm hand-color classification and mean shift tracking. In *Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems*, pages 119–124, Vancouver, BC, Canada, 2001.

[Kru91] M. Krueger. *Artificial Reality II.* Addison Wesley, Reading, MA, 1991.

[Kru93] M. Krueger. Environmental technology: Making the real world virtual. *Communications of the ACM*, 36:36–37, 1993.

[LB96] A. Leonardis and H. Bischof. Dealing with occlusions in the eigenspace approach. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 453–458, San Francisco, 1996.

[LF02] R. Lockton and R. Fitzgibbon. Real-time gesture recognition using deterministic boosting. In *Proc. British Machine Vision Conference (BMVC)*, pages 817–826, 2002.

[LK95] J. Lee and T. L. Kunii. Model-based analysis of hand posture. *IEEE Computer Graphics and Applications*, 15(5):77–86, 1995.

[LK99] H.-K. Lee and J. H. Kim. An HMM-based threshold model approach for gesture recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(10):961–973, 1999.

[LL01] I. Laptev and T. Lindeberg. Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features. In *Proc. Scale-Space'01*, volume 2106 of *Lecture Notes in Computer Science*, pages 63+, 2001.

[LTA95] A. Lanitis, T. Taylor, C. Cootes, and T. Ahmed. Automatic interpretation of human faces and hand gestures using flexible models. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 98–103, Zurich, 1995.

[LWH02] J. Lin, Y. Wu, and T. S. Huang. Capturing human hand motion in image sequences. In *Proc. IEEE workshop on Motion and Video Computing*, pages 99–104, 2002.

[LZ04] S. Li and H. Zhang. Multi-view face detection with floatboost. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(9):1112–1123, 2004.

[Mag95] C. Maggioni. GestureComputer - new ways of operating a computer. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 166–171, Zurich, Switzerland, 1995.

[MB99] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *Proc. International Conference on Computer Vision (ICCV)*, pages 572–578, Corfu, Greece, 1999.

[MC97] J. Martin and J. Crowley. An appearance-based approach to gesture-recognition. In *Int. Conf. on Image Analysis and Processing*, pages 340–347, Florence, Italy, 1997.

[MCA01] J. P. Mammen, S. Chaudhuri, and T. Agrawal. Simultaneous tracking of both hands by estimation of erroneous observations. In *Proc. British Machine Vision Conference (BMVC)*, Manchester, UK, September 2001.

[MDBP95] P. Maes, T. Darrell, B. Blumberg, and A. Pentland. The alive system: Full-body interaction with autonomous agents. In *Computer Animation Conference*, pages 11–18, Geneva, Switzerland, 1995.

[MDC98] J. Martin, V. Devin, and J. Crowley. Active hand tracking. In *IEEE Conference on Automatic Face and Gesture Recognition*, pages 573–578, Nara, Japan, 1998.

[MHP+01] W. MacLean, R. Herpers, C. Pantofaru, C. Wood, K. Derpanis, D. Topalovic, and J. Tsotsos. Fast hand gesture recognition for real-time teleconferencing applications. In *International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems*, pages 133–140, 2001.

[MI00] J. MacCormick and M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *Proc. European Conference on Computer Vision*, pages 3–19, 2000.

[ML97] P. Morguet and M. K. Lang. A universal HMM-based approach to image sequence classification. In *Proc. International Conference on Image Processing (ICIP)*, pages 146–149, 1997.

[MM95] D. J. Mapes and M. J. Moshell. A two-handed interface for object manipulation in virtual environments. *PRESENSE: Teleoperators and Virtual Environments*, 4(4):403–416, 1995.

[MM02] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *Proc. European Conference on Computer Vision*, volume 3, pages 666–680, Copenhagen, Denmark, 2002.

[MMR00] G. McAllister, S. McKenna, and I. Ricketts. Towards a non-contact driver-vehicle interface. In *Intelligent Transportation Systems*, pages 58–63, Dearborn, MI, 2000.

[MN95] H. Murase and S. Nayar. Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14:5–24, 1995.

[MP95] B. Moghaddam and A. Pentland. Maximum likelihood detection of faces and hands. In *Int. Conference on Automatic Face and Gesture Recognition*, pages 122–128, Zurich, Switzerland, 1995.

[MR92] A. Meyering and H. Ritter. Learning to recognize 3D-hand postures from perspective pixel images. In *Artificial Neural Networks II*, pages 821–824. Elsevier Science Publishers, 1992.

[MWSK02] A. Martinez, B. Wilbur, R. Shay, and A. Kak. Purdue rvl-slll asl database for automatic recogition of american sign language. In *International Conference on Multimodal Interfaces*, pages 167–172, 2002.

[NR98] C. Nolker and H. Ritter. Illumination independent recognition of deictic arm postures. In *Annual Conf. of the IEEE Industrial Electronics Society*, pages 2006–2011, Germany, 1998.

[OB04] E. Ong and R. Bowden. A boosted classifier tree for hand shape detection. In *Automatic Face and Gesture Recognition*, pages 889–894, 2004.

[OH97] C. Olson and D. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Transactions on Image Processing*, 6(1):103–113, 1997.

[OZ97] R. O'Hagan and A. Zelinsky. Finger Track - a robust and real-time gesture interface. In *Australian Joint Conference on Artificial Intelligence*, pages 475–484, Perth, Australia, November 1997.

[Pet99] N. Peterfreund. Robust tracking of position and velocity with Kalman snakes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 10(6):564–569, 1999.

[PHVG02] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Proc. European Conference on Computer Vision*, pages 661–675, Copenhagen, Denmark, May 2002.

[PSH96] V. Pavlovic, R. Sharma, and T. Huang. Gestural interface to a visual computing environment for molecular biologists. In *Int. Conf. Automatic Face and Gesture Recognition*, pages 30–35, Killington, VT, 1996.

[QMZ95] F. Quek, T. Mysliwiec, and M. Zhao. Finger mouse: A freehand pointing interface. In *IEEE Int. Workshop on Automatic Face and Gesture Recognition*, pages 372–377, Zurich, Switzerland, 1995.

[Que95] F. Quek. Eyes in the interface. *Image and Vision Computing*, 13(6):511–525, 1995.

[Que96] F. Quek. Unencumbered gesture interaction. *IEEE Multimedia*, 3(3):36–47, 1996.

[Que00] F. Quek. Gesture, speech, and gaze cues for discourse segmentation. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 247–254, 2000.

[QZ96] F. Quek and M. Zhao. Inductive learning in hand pose recognition. In *IEEE Automatic Face and Gesture Recognition*, pages 78–83, Killington, VT, 1996.

[RA97] S. Ranganath and K. Arun. Face recognition using transform features and neural networks. *Pattern Recognition*, 30(10):1615–1622, October 1997.

[RASS01] R. Rosales, V. Athitsos, L. Sigal, and S. Sclaroff. 3D hand pose reconstruction using specialized mappings. In *Proc. International Conference on Computer Vision (ICCV)*, pages 378–385, Vancouver, Canada, 2001.

[RG98] S. Raja and S. Gong. Tracking and segmenting people in varying lighting conditions using colour. In *Int. Conf. on Automatic Face and Gesture Recognition*, pages 228–233, Nara, Japan, 1998.

[RK94] J. Rehg and T. Kanade. Digiteyes: Vision-based hand tracking for human-computer interaction. In *Workshop on Motion of Non-Rigid and Articulated Bodies*, pages 16–24, Austin Texas, November 1994.

[RK95] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *Proc. International Conference on Computer Vision (ICCV)*, pages 612–617, 1995.

[RKE98] G. Rigoll, A. Kosmala, and S. Eickeler. High performance real-time gesture recognition using hidden Markov models. *Lecture Notes in Computer Science*, 1371:69–??, 1998.

[RKS96] G. Rigoll, A. Kosmala, and M. Schusterm. A new approach to video sequence recognition based on statistical methods. In *Proc. International Conference on Image Processing (ICIP)*, volume 3, pages 839–842, Lausanne, Switzerland, 1996.

[RMG98] Y. Raja, S. McKenna, and S. Gong. Colour model selection and adaptation in dynamic scenes. In *Proc. European Conference on Computer Vision*, pages 460–475, 1998.

[SC02] J. Sullivan and S. Carlsson. Recognizing and tracking human action. In *Proc. European Conference on Computer Vision*, volume 1, pages 629–644, Copenhagen, Denmark, 2002.

[Sch02] R. Schapire. The boosting approach to machine learning: An overview. In *MSRI Workshop on Nonlinear Estimation and Classification*, 2002.

[SEG99] C. Stauffer, W. Eric, and L. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 2246–2252, Ft. Collins, USA, June 1999.

[SF96] D. Saxe and R. Foulds. Toward robust skin identification in video images. In *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pages 379–384, 1996, 1996.

[She93] T. Sheridan. Space teleoperation through time delay: review and prognosis. *IEEE Transactions on Robotics and Automation*, 9(5):592–606, 1993.

[SHJ94a] J. Schlenzig, E. Hunter, and R. Jain. Recursive identication of gesture inputs using hidden markov models. In *IEEE Workshop on Applications of Computer Vision*, pages 187–194, Sarasota, FL, 1994.

[SHJ94b] J. Schlenzig, E. Hunter, and R. Jain. Vision based hand gesture interpretation using recursive estimation. In *Asilomar Conference Signals, Systems, and Computers*, 1994.

[SHWP07] H. Siegl, M. Hanheide, S. Wrede, and A. Pinz. An augmented reality human-computer interface for object localization in a cognitive vision system. *Image and Vision Computing*, 25:1895–1903, 2007.

[SK87] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, 4:519–524, March 1987.

[SK98] J. Segen and S. Kumar. Fast and accurate 3D gesture recognition interface. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 86–91, 1998.

[SKS01] N. Shimada, K. Kimura, and Y. Shirai. Real-time 3-d hand posture estimation based on 2-d appearance retrieval using monocular camera. In *Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems*, pages 23–30, Vancouver, Canada, 2001.

[SMC02] B. Stenger, R. Mendonca, and R. Cippola. Model-based 3D tracking of an articulated hand. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 126–133, Hawaii, 2002.

[SP95] T. Starner and A. Pentland. Visual recognition of american sign language using hidden markov models. In *IEEE International Symposium on Computer Vision*, 1995.

[SRG99] McKenna S., Y. Raja, and S. Gong. Tracking color objects using adaptive mixture models. *Image and Vision Computing*, 17(3):225–231, 1999.

[SS98] R. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. In *Annual Conf. on Computational Learning Theory*, pages 80–91, 1998.

[SS05] A. Shamaie and A. Sutherland. Hand tracking in bimanual movements. *Image and Vision Computing*, 23(13):1131–1149, 2005.

[SSA04] L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(7):862–877, 2004.

[SSK99] J. Segen and S. S. Kumar. Shadow gestures: 3D hand pose estimation using a single camera. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 479–485, 1999.

[SSKM98] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura. Hand gesture estimation and model refinement using monocular camera - ambiguity limitation by inequality constraints. In *IEEE Int. Conf. on Face and Gesture Recognition*, pages 268–273, Nara, Japan, 1998.

[ST05] L. Song and M. Takatsuka. Real-time 3D finger pointing for an augmented desk. In *Australasian conference on User interface*, volume 40, pages 99–108, Newcastle, Australia, 2005.

[STTC06] B. Stenger, A. Thayananthan, P. Torr, and R. Cipolla. Model-based hand tracking using a hierarchical bayesian filter. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(9):1372–1384,, September 2006.

[SWP98] T. Starner, J. Weaver, , and A. Pentland. Real-time american sign language recognition using desk and wearable computer-based video. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, 1998.

[TM96] J. Triesch and C. Malsburg. Robust classification of hand postures against complex background. In *IEEE Automatic Face and Gesture Recognition*, pages 170–175, Killington, VT, 1996.

[TP91] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Neuroscience*, 3(1):71–86, 1991.

[TPS03] C. Tomasi, S. Petrov, and A. Sastry. 3D tracking = classification + interpolation. In *Proc. International Conference on Computer Vision (ICCV)*, volume 2, pages 1441–1448, Nice, France, 2003.

[TSFA00] J. Terrillon, M. Shirazi, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 54–61, 2000.

[TSS02] N. Tanibata, N. Shimada, and Y. Shirai. Extraction of hand features for recognition of sign language words. In *Int. Conference on Vision Interface*, pages 391–398, 2002.

[TVdM98] J. Triesch and C. Von der Malsburg. A gesture interface for human-robot-interaction. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 546–551, Nara, Japan, April 1998. IEEE.

[UO97] A. Utsumi and J. Ohya. Direct manipulation interface using multiple cameras for hand gesture recognition. In *SIGGRAPH*, page 112, 1997.

[UO98] A. Utsumi and J. Ohya. Image segmentation for human tracking using sequential-image-based hierarchical adaptation. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 911–916, 1998.

[UO99] A. Utsumi and J. Ohya. Multiple-hand-gesture tracking using multiple cameras. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 473–478, Colorado, 1999.

[UV95] C. Uras and A. Verri. Hand gesture recognition from edge maps. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 116–121, Zurich, Switzerland, 1995.

[VD95] R. Vaillant and D. Darmon. Vision-based hand pose estimation. In *Int. Workshop on Automatic Face and Gesture Recognition*, pages 356–361, Zurich, Switzerland, 1995.

[VJ01] P. Viola and M. Jones. Robust real-time object detection. In *IEEE Workshop on Statistical and Computational Theories of Vision*, Vancouver, Canada, 2001.

[VJS03] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proc. International Conference on Computer Vision (ICCV)*, pages 734–741, 2003.

[VM98] C. Vogler and D. Metaxas. Asl recognition based on a coupling between HMMs and 3D motion analysis. In *Proc. International Conference on Computer Vision (ICCV)*, pages 363–369, 1998.

[VM99] C. Vogler and D. Metaxas. Toward scalability in asl recognition: Breaking down signs into phonemes. in gesture workshop. In *International Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction*, pages 211–224, 1999.

[VM01] C. Vogler and D. Metaxas. A framework for recognizing the simultaneous aspects of american sign language. *Computer Vision and Image Understanding*, 81(3):358–384, 2001.

[VPGB02] J. Vermaak, P. Perez, M. Gangnet, and A. Blake. Towards improved observation models for visual tracking: selective adaptation. In *Proc. European Conference on Computer Vision*, pages 645–660, Berlin, Germany, 2002.

[WADP97] C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. PFinder: Real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

[Wal95] M. Waldron. Isolated asl sign recognition sytem for deaf persons. *IEEE Transactions on Rehabilitation Engineering*, 3(3):261–271, 1995.

[WB95] A. Wilson and A. Bobick. Learning visual behavior for gesture analysis. In *IEEE Symposium on Computer Vision*, Coral Gables, FL, 1995.

[Wel93] P. Wellner. The digitaldesk calculator: Tangible manipulation on a desk top display. In *ACM Symposium on User Interface Software and Technology*, pages 27–33, 1993.

[WH00] Y. Wu and T. S. Huang. View-independent recognition of hand postures. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 84–94, Hilton Head Island, SC, 2000.

[WHSSdVL04] A. Wu, K. Hassan-Shafique, M . Shah, and N. da Vitoria Lobo. Virtual three-dimensional blackboard: Three-dimensional finger tracking with a single camera. *Applied Optics*, 43(2):379–390, 2004.

[WKSE02] J. Wachs, U. Kartoun, H. Stern, and Y. Edan. Real-time hand gesture telerobotic system. In *World Automation Congress*, volume 13, pages 403–409, Orlando, FL, 2002.

[WLH00] Y. Wu, Q. Liu, and T. Huang. An adaptive self-organizing color segmentation algorithm with application to robust real-time human hand localization. In *ACCV*, pages 1106–1111, Taipei, Taiwan, 2000.

[WLH01] Y. Wu, J. Lin, and T. Huang. Capturing natural hand articulation. In *Proc. International Conference on Computer Vision (ICCV)*, pages 426–432, Vancouver, Canada, July 2001.

[WO03] A. Wilson and N. Oliver. Gwindows: Robust stereo vision for gesture-based control of windows. In *International Conference on Multimodal Interfaces*, pages 211–218, Vancouver, Canada, 2003.

[WP97] C. Wren and A. Pentland. Dynamic models of human motion. In *IEEE Intl Conf. Automatic Face and Gesture Recognition*, pages 22–27, Nara, Japan, 1997.

[WTH99] Y. Wu and T. T. Huang. Capturing human hand motion: A divide-and-conquer approach. In *Proc. International Conference on Computer Vision (ICCV)*, pages 606–611, Greece, 1999.

[YA98] M. Yang and N. Ahuja. Detecting human faces in color images. In *Proc. International Conference on Image Processing (ICIP)*, pages 127–130, Piscataway, NJ, 1998.

[YAT02] M. H. Yang, N. Ahuja, and M. Tabb. Extraction of 2D motion trajectories and its application to hand gesture recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(8):1061–1074, August 2002.

[Yin03] M. Yin, X.and Xie. Estimation of the fundamental matrix from uncalibrated stereo hand images for 3D hand gesture recognition. *Pattern Recognition*, 36(3):567–584, 2003.

[YK04] S. M. Yoon and H. Kim. Real-time multiple people detection using skin color, motion and appearance information. In *Proc. IEEE International Workshop on Robot and Human Interactive Communication (ROMAN)*, pages 331–334, Kurashiki, Okayama Japan, September 2004.

[YLW98] J. Yang, W. Lu, and A. Waibel. Skin-color modeling and adaptation. In *ACCV*, pages 687–694, 1998, 1998.

[YOI92] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden markov model. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 379–385, 1992.

[YSA95] Q. Yuan, S. Sclaroff, and V. Athitsos. Automatic 2D hand tracking in video sequences. In *IEEE Workshop on Applications of Computer Vision*, pages 250–256, 1995.

[ZH05] J. P. Zhou and J. Hoang. Real time robust human detection and tracking system. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages III: 149–149, 2005.

[ZNG+04] M. Zobl, R. Nieschulz, M. Geiger, M. Lang, and G. Rigoll. Gesture components for natural interaction with in-car devices. In *International Gesture Workshop in Gesture-Based Communication in Human-Computer Interaction*, pages 448–459, Gif-sur-Yvette, France, 2004.

[ZPD+97] M. Zeller, C. Phillips, A. Dalke, W. Humphrey, K. Schulten, S. Huang, I. Pavlovic, Y. Zhao, Z. Lo, S. Chu, and R. Sharma. A visual computing environment for very large scale biomolecular modeling. In *IEEE Int. Conf. Application-Specific Systems, Architectures and Processors*, pages 3–12, Zurich, Switzerland, 1997.

[ZYW00] X. Zhu, J. Yang, and A. Waibel. Segmenting hands of arbitrary color. In *Proc. International Conference on Automatic Face and Gesture Recognition (FG)*, pages 446–455, Grenoble, France, March 2000.