A Prototype System for Computer Vision Based Human Computer Interaction*

Lars Bretzner,[1] *Ivan Laptev,*[1] *Tony Lindeberg,*[1]
Sören Lenman,[2] *Yngve Sundblad*[2]

[1]Computational Vision and Active Perception Laboratory (CVAP)
[2]Center for User-Oriented IT-Design (CID)
Department of Numerical Analysis and Computing Science
KTH (Royal Institute of Technology)
S-100 44 Stockholm, Sweden.

Technical report ISRN KTH/NA/P–01/09–SE

1 Introduction

With the development of information technology in our society, we can expect that computer systems to a larger extent will be embedded into our environment. These environments will impose needs for new types of human-computer-interaction, with interfaces that are natural and easy to use. In particular, the ability to interact with computerized equipment without need for special external equipment is attractive.

Today, the keyboard, the mouse and the remote control are used as the main interfaces for transferring information and commands to computerized equipment. In some applications involving three-dimensional information, such as visualization, computer games and control of robots, other interfaces based on trackballs, joysticks and datagloves are being used. In our daily life, however, we humans use our vision and hearing as main sources of information about our environment. Therefore, one may ask to what extent it would be possible to develop computerized equipment able to communicate with humans in a similar way, by understanding visual and auditive input.

Perceptual interfaces based on speech have already started to find a number of commercial and technical applications. For examples, systems are now available where speech commands can be use for dialing numbers in cellular

*The support from the Swedish Research Council for Engineering Sciences, TFR, and the Swedish National Board for Industrial and Technical Development, NUTEK, is gratefully acknowledged. On-line video clip demos can be viewed from `http://www.nada.kth.se/cvap/gvmdi`, and an on-line version of this manuscript can be fetched from `http://www.nada.kth.se/cvap/abstracts/cvap251.html`.

phones or for making ticket reservations. Concerning visual input, the processing power of computers has reached a point where real-time processing of visual information is possible with common workstations.

The purpose of this article is to describe ongoing work in developing new perceptual interfaces with emphasis on commands expressed as hand gestures. Examples of applications of hand gesture analysis include:

- Control of consumer electronics
- Interaction with visualization systems
- Control of mechanical systems
- Computer games

Potential advantages of using visual input in this context are that visual information makes it possible to communicate with computerized equipment at a distance, without need for physical contact with the equipment that is to be controlled. Moreover, the user should be able to control the equipment without need for specialized external devices, such as a remote control.

2 Control by hand gestures

Figure 1 shows an illustration of a type of scenario we are interested in. The user is in front of a camera connected to a computer. The camera follows the movements of the hand, and performs actions depending on the state and the motion of the hand. Three basic types of hand gestures can be identified in such a situation:

- A *static hand posture* implies that the hand is held in a fixed state during a certain period of time, during which the system recognizes the state given a predefined set of states. Examples of interpretations that are possible

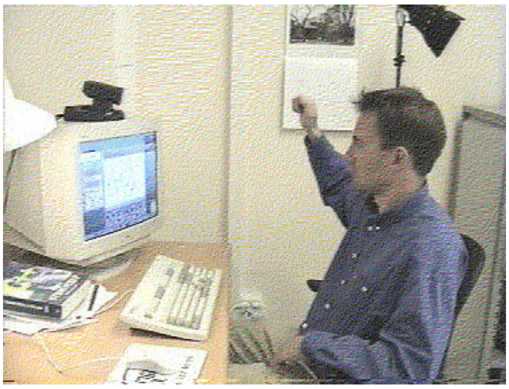


Figure 1: Example of a simple situation where the user controls actions on a screen using hand gestures. In this application, the position of the cursor is controlled by the motion of the hand, and the user can induce a click by changing the hand posture.

include on or off for a TV, start or stop for a video recorder, or a choice between different modes for a command involving motion.

- A *quantitative hand motion* means that the two-dimensional or the three-dimensional motion of the hand is measured, and the estimated motion parameters (translations and rotations) are being used for controlling the motion of other computerized equipment, such as visualization parameters for displaying a three-dimensional object, the volume of a TV or the motion of robot.

- A *qualitative hand motion* means that the hand moves according to a predefined motion pattern (a trajectory in space-time) and that the motion pattern is recognized from a predefined set of motion patterns. Examples of interpretations include letters (the Palm Pilot sign language) or control of consumer electronics in a similar manner as for static hand postures.

3 A prototype scenario

To be able to test computer-vision-based human-computer-interaction in practice, we developed a prototype test bed system, where the user can control a TV set and a lamp using the following types of hand postures:

- Three open fingers (figure 2(a)) toggle the TV on or off.

- Two open fingers (figure 2(b-c)) change the channel of the TV. With the index finger pointing to one side, the next TV channel is selected, while the previous channel is selected if the index finger points upwards.

- Five open fingers (figure 2(d)) toggle the lamp on or off.

Figure 3 shows a few snapshots from a demonstration, where a user controls equipment in the environment in this way. In figures 3(a)–(b) a user turns on the lamp, in figures 3(c)–(d) he turns on the TV set, and in figures 3(e)–(f) he switches the TV set to a new channel. All steps in this demonstration have

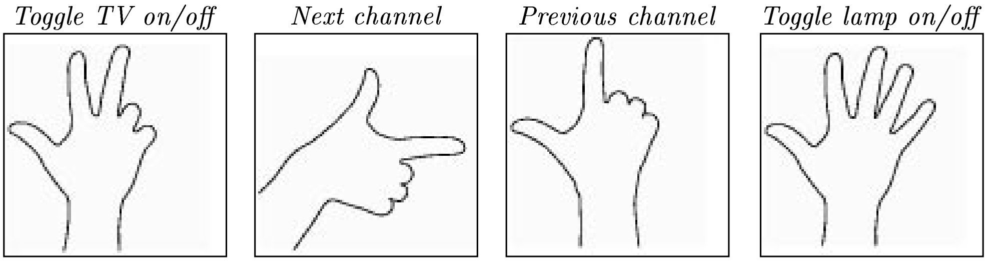
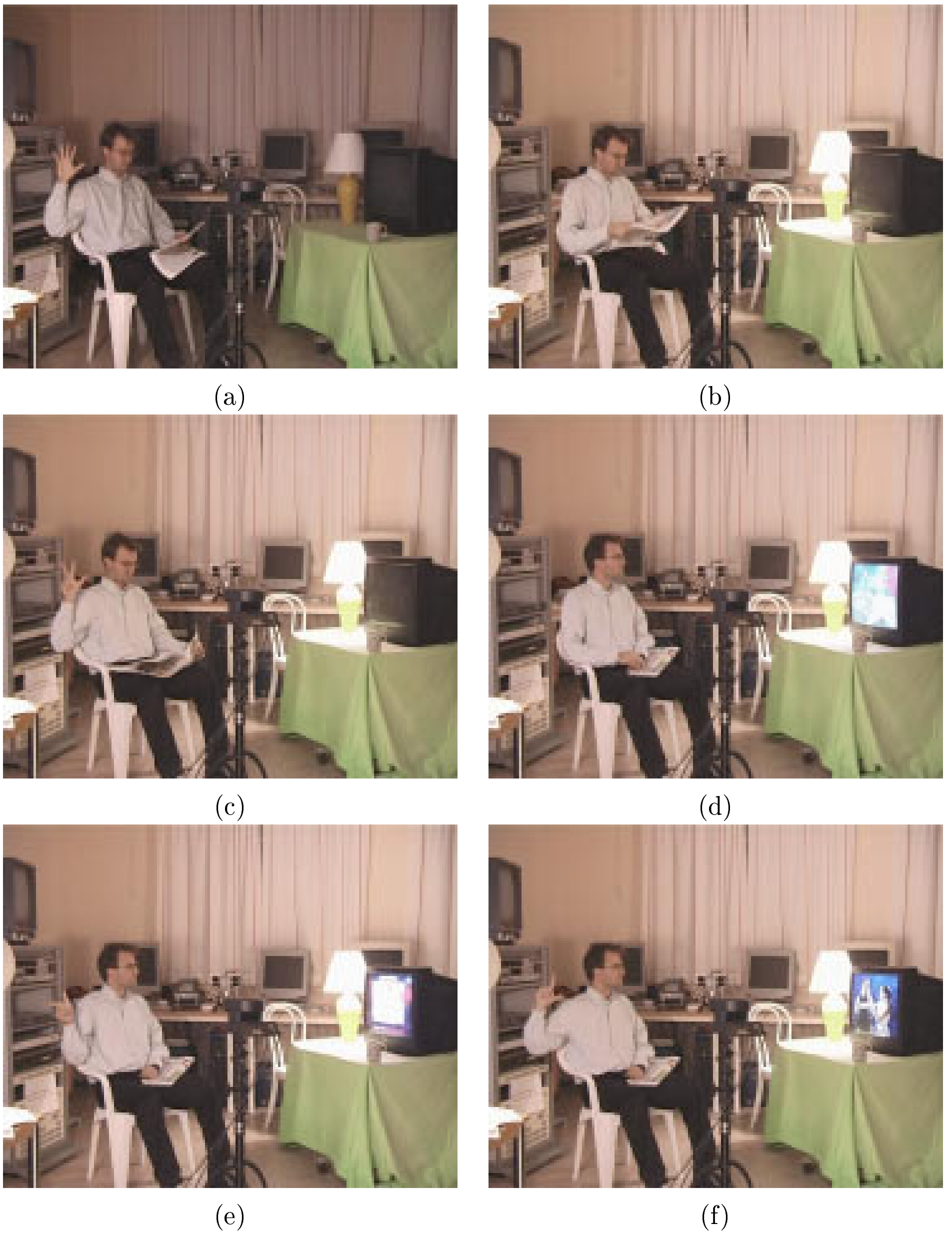


Figure 2: Hand postures controlling a prototype scenario: (a) a hand with three open fingers toggles the TV on or off, (b) a hand with two open fingers and the index finger pointing to one side selects the next TV channel, (c) a hand with two open fingers and the index finger pointing upwards selects the previous TV channel, (d) a hand with five open fingers toggles the lamp on or off.



(a) (b)

(c) (d)

(e) (f)

Figure 3: A few snapshots from a scenario where a user enters a room and turns on the lamp (a)-(b), turns on the TV set (c)-(d) and switches to a new TV channel (e)-(f).

been performed in real-time and during continuous operation of the prototype system described in next section.

4 A prototype system

To track and recognize hands in multiple states, we have developed a system based on a combination of shape and colour information. At an overview level, the system consists of the following functionalities (see figure 4):

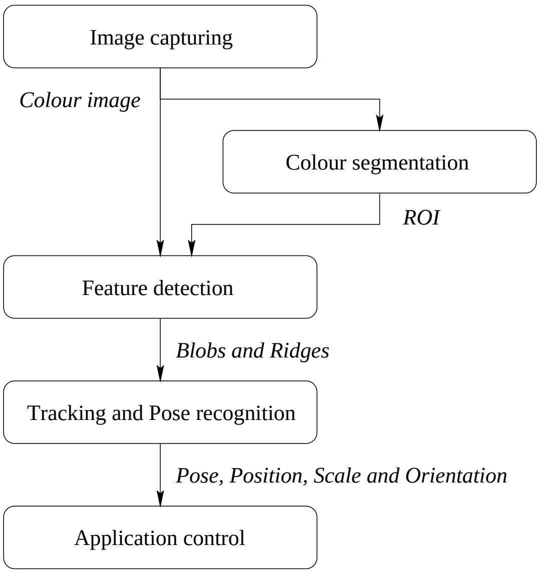


Figure 4: Overview of the main components of the prototype system for detecting and recognizing hand gestures, and using this information for controlling consumer electronics.

The image information from the camera is grabbed at frame rate, the colour images are converted from RGB format to a new colour space that separates the intensity and chromaticity components of the colour data. In the colour images, colour feature detection is performed, which results in a set of image features that can be matched to a model. Moreover, a complementary comparison between actual colour and skin colour is performed to identify regions that are more likely to contain hands. Based on the detected image features and the computed skin colour similarity, comparison with a set of object hypotheses is performed using a statistical approach referred to as particle filtering or condensation. The most likely hand posture is estimated, as well as the position, size and orientation of the hand. This recognized gesture information is bound to different actions relative to the environment, and these actions are carried under the control of the gesture recognition system. In this way, the gesture recognition system provides a medium by which the user can control different types of equipment in his environment. Appendix A gives a more detailed description of the algorithms and computational modules in the system.

5 Related works

The problem of hand gesture analysis has received increased attention recent years. Early work of using hand gestures for television control was presented by (Freeman & Weissman 1995) using normalized correlation; see also (Kuch & Huang 1995, Pavlovic et al. 1997, Maggioni & Kämmerer 1998, Cipolla & Pentland 1998) for related works. Some approaches consider elaborated 3-D hand models (Regh & Kanade 1995), while others use colour markers to simplify feature detection (Cipolla et al. 1993). Appearance-based models for hand tracking and sign recognition were used by (Cui & Weng 1996), while (Heap & Hogg 1998, MacCormick & Isard 2000) tracked silhouettes of hands. Graph-like and feature-based hand models have been proposed by (Triesch & von der Malsburg 1996) for sign recognition and in (Bretzner & Lindeberg 1998) for tracking and estimating 3-D rotations of a hand.

The use of a hierarchical hand model continues along the works by (Crowley & Sanderson 1987) who extracted peaks from a Laplacian pyramid of an image and linked them into a tree structure with respect to resolution, (Lindeberg 1993) who constructed scale-space primal sketch with an explicit encoding of blob-like structures in scale space as well as the relations between these, (Triesch & von der Malsburg 1996) who used elastic graphs to represent hands in different postures with local jets of Gabor filters computed at each vertex, (Lindeberg 1998) who performed feature detection with automatic scale selection by detecting local extrema of normalized differential entities with respect to scale, (Shokoufandeh et al. 1999) who detected maxima in a multi-scale wavelet transform, as well as (Bretzner & Lindeberg 1999), who computed multi-scale blob and ridge features and defined explicit qualitative relations between these features. The use of chromaticity as a primary cue for detecting skin coloured regions was first proposed by (Fleck et al. 1996).

Our implementation of particle filtering largely follows the traditional approaches for condensation as presented by (Isard & Blake 1996, Black & Jepson 1998, Sidenbladh et al. 2000, Deutscher et al. 2000) and others. Using the hierarchical multi-scale structure of the hand models, however, we adapted the layered sampling approach (Sullivan et al. 1999) and used a coarse-to-fine search strategy to improve the computational efficiency, here, by a factor of two.

The proposed approach is based on several of these works and is novel in the respect that it combines a hierarchical object model with image features at multiple scales and particle filtering for robust tracking and recognition. For more details about the algorithmic aspects underlying the tracking and recognition components in the current system, see (Laptev & Lindeberg 2000).

6 The CVAP-CID collaboration

The work is carried out as a collaboration project between the Computational Vision and Active Perception Laboratory (CVAP) and the Center for User-Oriented IT-Design at KTH, where CVAP provides expertise on computer vision, while CID provides expertise on human-computer-interaction.

In the development of new forms of human computer interfaces, it is of central importance that user studies are being carried out and that the interaction is tested in prototype systems as early as possible. Computer vision algorithms for gesture recognition will be developed by CVAP, and will be used in prototype systems in scenarios defined in collaboration with CID. User studies for these scenarios will then be performed and be developed by CID, to guide further developments.

References

Black, M. & Jepson, A. (1998), A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gestures and expressions, *in* 'Fifth European Conference on Computer Vision', Freiburg, Germany, pp. 909–924.

Bretzner, L. & Lindeberg, T. (1998), Use your hand as a 3-D mouse or relative orientation from extended sequences of sparse point and line correspondences using the affine trifocal tensor, *in* H. Burkhardt & B. Neumann, eds, 'Fifth European Conference on Computer Vision', Vol. 1406 of *Lecture Notes in Computer Science*, Springer Verlag, Berlin, Freiburg, Germany, pp. 141–157.

Bretzner, L. & Lindeberg, T. (1999), Qualitative multi-scale feature hierarchies for object tracking, *in* O. F. O. M. Nielsen, P. Johansen & J. Weickert, eds, 'Proc. 2nd International Conference on Scale-Space Theories in Computer Vision', Vol. 1682, Springer Verlag, Corfu, Greece, pp. 117–128.

Cipolla, R., Okamoto, Y. & Kuno, Y. (1993), Robust structure from motion using motion parallax, *in* 'Fourth International Conference on Computer Vision', Berlin, Germany, pp. 374–382.

Cipolla, R. & Pentland, A., eds (1998), *Computer vision for human-computer interaction*, Cambridge University Press, Cambridge, U.K.

Crowley, J. & Sanderson, A. (1987), 'Multiple resolution representation and probabilistic matching of 2-d gray-scale shape', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **9**(1), 113–121.

Cui, Y. & Weng, J. (1996), View-based hand segmentation and hand-sequence recognition with complex backgrounds, *in* '13th International Conference on Pattern Recognition', Vienna, Austria, pp. 617–621.

Deutscher, J., Blake, A. & Reid, I. (2000), Articulated body motion capture by annealed particle filtering, *in* 'CVPR'2000', Hilton Head, SC, pp. II:126–133.

Fleck, M., Forsyth, D. & Bregler, C. (1996), Finding naked people, *in* 'Fourth European Conference on Computer Vision', Cambridge, UK, pp. II:593–602.

Freeman, W. T. & Weissman, C. D. (1995), Television control by hand gestures, *in* 'Proc. Int. Conf. on Face and Gesture Recognition', Zurich, Switzerland.

Heap, T. & Hogg, D. (1998), Wormholes in shape space: Tracking through discontinuous changes in shape, *in* 'Sixth International Conference on Computer Vision', Bombay, India, pp. 344–349.

Isard, M. & Blake, A. (1996), Contour tracking by stochastic propagation of conditional density, *in* 'Fourth European Conference on Computer Vision', Cambridge, UK, pp. I:343–356.

Kuch, J. J. & Huang, T. S. (1995), Vision based hand modelling and tracking for virtual teleconferencing and telecollaboration, *in* 'Proc. 5th International Conference on Computer Vision', Cambridge, MA, pp. 666–671.

Laptev, I. & Lindeberg, T. (2000), Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features, Technical Report ISRN KTH/NA/P--00/12--SE, Dept. of Numerical Analysis and Computing Science, KTH, Stockholm, Sweden.

Lindeberg, T. (1993), 'Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention', *International Journal of Computer Vision* **11**(3), 283–318.

Lindeberg, T. (1998), 'Feature detection with automatic scale selection', *International Journal of Computer Vision* **30**(2), 77–116.

MacCormick, J. & Isard, M. (2000), Partitioned sampling, articulated objects, and interface-quality hand tracking, *in* 'Sixth European Conference on Computer Vision', Dublin, Ireland, pp. II:3–19.

Maggioni, C. & Kämmerer, B. (1998), Gesturecomputer-history, design and applications, *in* R. Cipolla & A. Pentland, eds, 'Computer vision for human-computer interaction', Cambridge University Press, Cambridge, U.K., pp. 23–52.

Pavlovic, V. I., Sharma, R. & Huang, T. S. (1997), 'Visual interpretation of hand gestures for human-computer interaction: A review', *IEEE Trans. Pattern Analysis and Machine Intell.* **19**(7), 677–694.

Regh, J. M. & Kanade, T. (1995), Model-based tracking of self-occluding articulated objects, *in* 'Fifth International Conference on Computer Vision', Cambridge, MA, pp. 612–617.

Shokoufandeh, A., Marsic, I. & Dickinson, S. (1999), 'View-based object recognition using saliency maps', *Image and Vision Computing* **17**(5/6), 445–460.

Sidenbladh, H., Black, M. & Fleet, D. (2000), Stochastic tracking of 3d human figures using 2d image motion, *in* 'Sixth European Conference on Computer Vision', Dublin, Ireland, pp. II:702–718.

Sullivan, J., Blake, A., Isard, M. & MacCormick, J. (1999), Object localization by bayesian correlation, *in* 'Seventh International Conference on Computer Vision', Corfu, Greece, pp. 1068–1075.

Triesch, J. & von der Malsburg, C. (1996), Robust classification of hand postures against complex background, *in* 'Proc. Int. Conf. on Face and Gesture Recognition', Killington, Vermont, pp. 170–175.

A Computational modules in the prototype system

This appendix gives a more detailed description of the algorithms underlying the different computational modules in the prototype system for hand gesture recognition outlined in section 4. In contrast to the main text, this presentation assumes knowledge about computer vision.

A.1 Shape cues

For each image, a set of blob and ridge features is detected. The idea is that the palm of the hand gives rise to a blob at a coarse scale, each one of the fingers gives rise to a ridge at a finer scale, and each finger tip gives rise to a fine scale blob. Figure 5 shows an example of such image features computed from an image.

A.1.1 Colour feature detection

Technically, this feature detection step is based on the following computational steps. The input colour image is transformed from the RGB colour space to a

Iuv colour space according to:

$$I = \frac{R+G+B}{3} \tag{1}$$

$$u = R - G \tag{2}$$

$$v = G - B \tag{3}$$

A scale-space representation is computed of each colour channel f_i by convolution with Gaussian kernels $g(\cdot;\ t)$ of different variance t, $C_i(\cdot;\ t) = g(\cdot;\ t) * f_i(\cdot)$ and the following normalized differential expressions are computed and summed up over the channels at each scale:

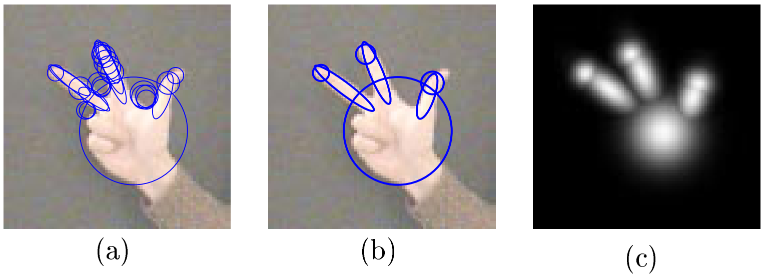
$$\mathcal{B}_{norm} C = \sum_C t^2 (\partial_{xx} C_i + \partial_{yy} C_i)^2 \tag{4}$$

$$\mathcal{R}_{norm} C = \sum_C t^{3/2} (\partial_{xx} C_i - \partial_{yy} C_i)^2 + 4(\partial_{xy} C_i)^2 \tag{5}$$

Then, scale-space maxima of these normalized differential entities are detected, i.e., points at which $\mathcal{B}_{norm}$ and $\mathcal{R}_{norm}$ assume normalized maxima with respect to space and scale. At each scale-space maximum $(x;\ t)$ a second-moment matrix

$$\nu = \sum_i \int_{\eta \in \mathbb{R}^2} \begin{pmatrix} (\partial_x C_i)^2 & (\partial_x LCi)(\partial_y C_i) \\ (\partial_x C_i)(\partial_y C_i) & (\partial_y C_i)^2 \end{pmatrix} g(\eta; s_{int})\, d\eta \tag{6}$$

is computed at integration scale s_{int} proportional to the scale of the detected image features. To allow for the computational efficiency needed to reach real-time performance, all the computations in the feature detection step have been implemented within a pyramid framework. Figure 5 shows such features, illustrated by ellipses centered at x and with covariance matrix $\Sigma = t\nu_{norm}$, where $\nu_{norm} = \nu/\lambda_{min}$ and λ_{min} is the smallest eigenvalue of ν.



(a) (b) (c)

Figure 5: The result of computing blob features and ridge features from an image of a hand. (a) circles and ellipses corresponding to the significant blob and ridge features extracted from an image of a hand; (b) selected image features corresponding to the palm, the fingers and the finger tips of a hand; (c) a mixture of Gaussian kernels associated with blob and ridge features illustrating how the selected image features capture the essential structure of a hand.

A.2 Hand model

As mentioned above, an image of a hand can be expected to give rise to blob and ridge features corresponding to the fingers of the hand. These image structures, together with information about their relative orientation, position and scale, can be used for defining a simple but discriminative view-based model of a hand. Thus, we represent a hand by a set of blob and ridge features as illustrated in figure 6, and define different states, depending on the number of open fingers.

To model translations, rotations and scaling transformations of the hand, we define a parameter vector $X = (x, y, s, \alpha, l)$, which describes the global position (x, y), the size s, and the orientation α of the hand in the image, together with its discrete state $l = 1 \ldots 5$. The vector X uniquely identifies the hand configuration in the image and estimation of X from image sequences corresponds to simultaneous hand tracking and recognition.

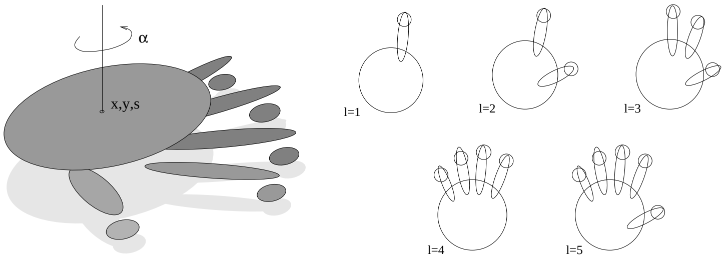


Figure 6: Feature-based hand models in different states. The circles and ellipses correspond to blob and ridge features. When aligning models to images, the features are translated, rotated and scaled according to the parameter vector X.

A.3 Skin colour

When tracking human faces and hands in images, the use of skin colour has been demonstrated to be a powerful cue. In this work, we explore similarity to skin colour in two ways:

- For defining candidate regions (masks) for searching for hands.
- For computing a probabilistic measure of any pixel being skin coloured.

Histogram-based computation of skin coloured search regions. To delimit regions in the image for searching for hands, an adaptive histogram analysis of colour information is performed. For every image, a histogram is computed for the chromatic (u, v)-components of the colour space. In this (u, v)-space a coarse search region has been defined, where skin coloured regions are likely to be. Within this region, blob detection is performed, and the blob most likely to correspond to skin colour is selected. The support region of this blob in colour space is backprojected into the image domain, which results in a number of skin coloured regions. Figure 7 shows an example of a region-of-interest interest computed in this way, which are used as a guide for subsequent processing.

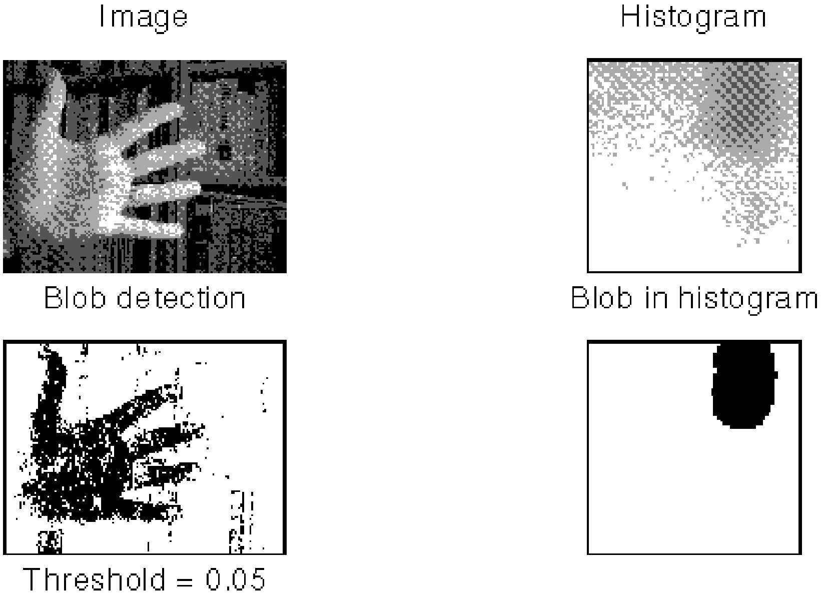


Figure 7: To delimit the regions in space where to perform recognition of hand gestures, an initial computation of regions of interest is carried out, based on adaptive histogram analysis. This illustration shows the behaviour of the histogram based colour analysis for a detail of a hand. In the system, however, the algorithm operates on overview images. (a) original image, (b) histogram over chromatic information, (c) backprojected histogram blob giving a hand mask, (d) results of blob detection in the histogram.

Probabilistic prior on skin colour. For exploring colour information in this context, we compute a probabilistic colour prior in the following way:

- Hands were segmented manually from the background for approximately 30 images, and two-dimensional histograms over the chromatic information (u, v) were accumulated for skin regions and background.

- These histograms were summed up and normalized to unit mass.

- Given these training data, the probability of any measured image point with colour values (u, v) being skin colour was estimated as

$$p_{skin}(u,v) = \frac{\max(0, a\, H_{skin}(u,v) - H_{bg}(u,v))}{\sum_{u,v} \max(0, a\, H_{skin}(u,v) - H_{bg}(u,v))}, \qquad (7)$$

For each hand model, this prior is evaluated at a number of image positions, given by the positions of the image features. Figure 8 shows the result of computing a map of this prior for an image with a hand.

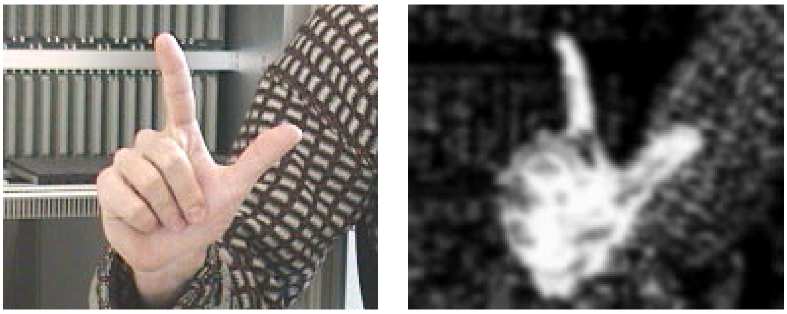


Figure 8: Illustration of the effect of the colour prior. (a) original image, (b) map of the the probability of skin colour at every image point.

A.4 Hand tracking and hand posture recognition

Tracking and recognition of a set of object models in time-dependent images can be formulated as the maximization of a posterior probability distribution over model parameters, given a sequence of input images. To estimate the states of object models in this respect, we follow the approach of particle filtering to propagate hypotheses over time.

A.4.1 Particle filtering

Particle filters aim at estimating and propagating the posterior probability distribution $p(X_t, Y_t|\tilde{\mathcal{I}}_t)$ over time, where X_t and Y_t are static and dynamic model parameters and $\tilde{\mathcal{I}}_t$ denotes the observations up to time t. Using Bayes rule, the posterior at time t can be evaluated according to

$$p(X_t, Y_t|\tilde{\mathcal{I}}_t) = k\, p(\mathcal{I}_t|X_t, Y_t)\, p(X_t, Y_t|\tilde{\mathcal{I}}_{t-1}), \tag{8}$$

where k is a normalization constant that does not depend on variables X_t,Y_t. The term $p(\mathcal{I}_t|X_t, Y_t)$ denotes the likelihood that a model configuration X_t, Y_t gives rise to the image $\mathcal{I}_t$. Using a first-order Markov assumption, the dependence on observations before time $t-1$ can be removed and the model prior $p(X_t, Y_t|\tilde{\mathcal{I}}_{t-1})$ can be evaluated using a posterior from a previous time step and the distribution for model dynamics according to

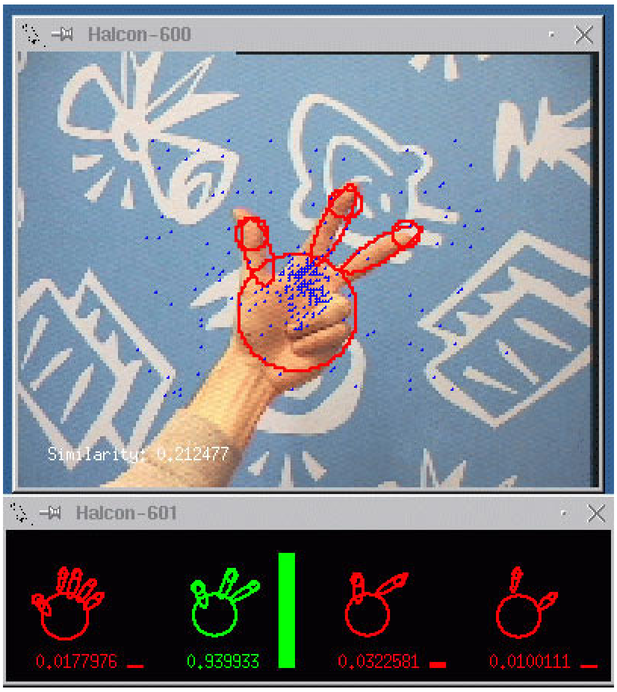
$$p(X_t, Y_t|\tilde{\mathcal{I}}_{t-1}) = \int p(X_t, Y_t|X_{t-1}, Y_{t-1})\, p(X_{t-1}, Y_{t-1}|\tilde{\mathcal{I}}_{t-1})\, dX_{t-1}\, dY_{t-1}. \tag{9}$$

Since the likelihood function is usually multi-modal and cannot be expressed in closed form, the approach of particle filtering is to approximate the posterior distribution using N particles, weighted according to their likelihoods $p(\mathcal{I}_t|X_t, Y_t)$. The posterior for a new time moment is then computed by populating the particles with high weights and predicting them according to their dynamic model $p(X_t, Y_t|X_{t-1}, Y_{t-1})$.

A.4.2 Hand tracking and posture recognition

To use particle filtering for tracking and recognition of hierarchical hand models, we let the state variable X denote the position (x, y), the size s, the orientation α and the posture l of the hand model, i.e., $X = (x, y, s, \alpha, l)$, while Y denotes the time derivatives of the first four variables, i.e., $Y_t = (\dot{x}, \dot{y}, \dot{s}, \dot{\alpha})$. Then, we assume that the likelihood $p(\mathcal{I}_t|X_t, Y_t)$ does not explicitly depend on Y_t, and approximate $p(\mathcal{I}_t|X_t)$ by evaluating $p(\mathcal{F}^d|\mathcal{F}^m)$ for each particle according to (15). Concerning the dynamics $p(X_{t-1}, Y_{t-1}|\tilde{\mathcal{I}}_{t-1})$ of the hand model, a constant velocity model is adopted, where deviations from the constant velocity assumption are modelled by additive Brownian motion, from which the distribution $p(X_t, Y_t|X_{t-1}, Y_{t-1})$ is computed. To capture changes in hand postures, the state parameter l is allowed to vary randomly for 30 % of the particles at each time step.

When the tracking is started, all particles are first distributed uniformly over the parameter spaces X and Y. After each time step of particle filtering, the best hypothesis of a hand is estimated, by first choosing the most likely hand posture and then computing the mean of $p(X_t, l_t, Y_t|\tilde{\mathcal{I}}_t)$ for that posture. Hand posture number i is chosen if $w_i = \max_j(w_j)$, $j = 1, \ldots, 5$, where w_j is the sum of the weights of all particles with state j. Then, the continuous parameters are estimated by computing a weighted mean of all the particles in state i. Figure 9 shows an example of model selection performed in this way.



Similarity: 0.212477

0.0177976

0.939933

0.0322581

0.0100111

Figure 9: At every time moment, the hand tracker based on particle filtering evaluates and compares a set of object hypothesis. From these hypothesis, which represent the probability distribution of the object, the most likely object state is estimated.

A.4.3 Model likelihood

To compute the likelihood of an object model given a set of image features, we represent each feature in the model and the data by a Gaussian kernel $g(x, \mu, \Sigma)$ having the same mean and covariance as the image features. Thus, the model and the data are represented by Gaussian mixture models according to

$$G^m = \sum_i^{N^m} \bar{g}(x, \mu_i^m, \Sigma_i^m), \quad G^d = \sum_i^{N^d} \bar{g}(x, \mu_i^d, \Sigma_i^d). \tag{10}$$

where $\bar{g}(x, \mu, \Sigma) = h(\Sigma)\, g(x, \mu, \Sigma)$ and the normalization factor is chosen as $h(\Sigma) = \sqrt[4]{\det(\Sigma)}$ to give scale invariance. To compare the model with the data, we integrate the square difference between their associated Gaussian mixture models

$$\Phi(\mathcal{F}^m, \mathcal{F}^d) = \int_{\mathbb{R}^2} (G^m - G^d)^2 \, dx. \tag{11}$$

which after a few approximations can be simplified to

$$\Phi(\mathcal{F}^m, \mathcal{F}^d) \approx \sum_{i=1}^{N^m} \phi(F_i^m, F_{k_i}^d) + \frac{N^d - N^m}{4\pi}, \tag{12}$$

where $\phi(F_i^m, F_{k_i}^d)$ denotes the square difference between a Gaussian representative of a model feature F_i^m and its nearest data feature $F_{k_i}^d$. An important property of this penalty term is that it allows for simultaneous localization and recognition of the object.

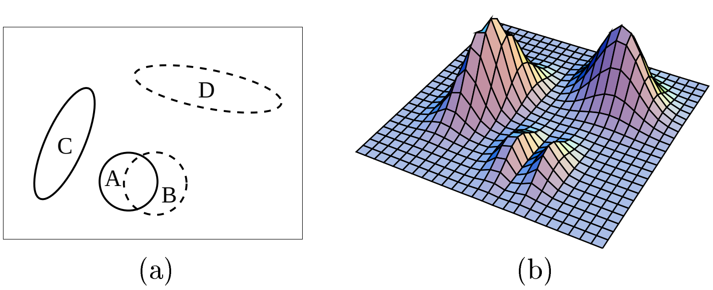


Figure 10: Two model features (solid ellipses) and two data features (dashed ellipses) in (a) are compared by evaluating the square difference of associated Gaussian functions. While the overlapping model (A) and the data (B) features cancel each other, the mismatched features (C and D) increase the square difference in (b).

After a few calculations, it can be shown that (12) can be expressed in closed form as

$$\phi(F_1, F_2) = \frac{1}{2\pi} - C\, \frac{\sqrt[4]{\det(\Sigma_1^{-1}) \det(\Sigma_2^{-1})}}{\pi \sqrt{\det(\Sigma_1^{-1} + \Sigma_2^{-1})}}. \tag{13}$$

where

$$C = \exp\left(-\tfrac{1}{2}(\mu_1^T \Sigma_1^{-1} \mu_1 + \mu_2^T \Sigma_2^{-1} \mu_2 - (\mu_1^T \Sigma_1^{-1} + \mu_2^T \Sigma_2^{-1})\hat{\mu})\right) \tag{14}$$

Given these entities, the likelihood of a model feature is then estimated by

$$p(\mathcal{F}^d | \mathcal{F}^m) = e^{-\Phi(\mathcal{F}^m, \mathcal{F}^d)/2\sigma^2}, \tag{15}$$

where $\sigma = 10^{-2}$ controls the sharpness of the likelihood function, and this entity is multiplied by the prior $p_{skin}(u, v)$ on skin colour.