Head Pose Estimation Using Stereo Vision For Human-Robot Interaction

Edgar Seemann Kai Nickel Rainer Stiefelhagen

Interactive Systems Labs
Universität Karlsruhe (TH)
Germany

Abstract

In this paper we present a method for estimating a person's head pose with a stereo camera. Our approach focuses on the application of human-robot interaction, where people may be further away from the camera and move freely around in a room. We show that depth information acquired from a stereo camera not only helps improving the accuracy of the pose estimation, but also improves the robustness of the system when the lighting conditions change.

The estimation is based on neural networks, which are trained to compute the head pose from grayscale and disparity images of the stereo camera. It can handle pan and tilt rotations from -90° to $+90^\circ$. Our system doesn't require any manual initialization and doesn't suffer from drift during an image sequence. Moreover the system is capable of real-time processing.

1 Introduction

Advancing human-robot interaction has been an active research field in recent years [10, 11, 12, 13, 14]. A major challenge is the tracking and interpretation of human behaviors in video data, since it is essential for enabling natural human-robot interaction.

In order to fully understand what a user does or intends to do, a robot should be capable of detecting and understanding many of the communicative cues used by humans. This involves in particular the recognition and interpretation of human faces. In interaction between people faces are continuously used to signal interest, emotion and direction of attention. Monitoring such information therefore can be used to make interaction with a robot more natural and intuitive.

Monitoring a person's head orientation is an important step towards building better human-robot interfaces. Since head orientation is related to a person's direction of attention, it can give us useful information about the objects or persons with which a user is interacting. It can furthermore be used to help a robot decide whether he was addressed by a person or not [16].

In this work we propose a method for estimating a person's head pose from an image pair of a stereo camera. Our approach consists of four steps. In the first step the depth information is calculated from the image pairs. During step two the user's face is detected and extracted from the image. Step three converts the resulting three dimensional head model into a input pattern for a neural network. Finally, a neural network is used to estimate the user's current head orientation.

All components of our system work fully automatic and do not require any manual initialization.

For our system, depth information is crucial in many ways. Firstly, depth information improves the accuracy of the pose estimation. Moreover depth information makes it easier to track the user's face in the image sequence. And finally, depth information improves robustness of the system when lighting conditions change.

In our tests we used a Videre Design stereo camera and its SVS library [15] to compute the depth information for the pixels. For the analysis of our pose estimates, we compared the estimated angles to those of a magnetic sensor (Flock of Birds).

We evaluated our system on both known and unknown users. Moreover we analyzed the performance in different application scenarios and under different lighting conditions.

The remainder of the paper is organized as follows: In section 1.1 related work is presented. Section 2 covers our newly developed head pose estimation method. First, the head detection and extraction technique are described, then we show how the neural network patterns are calculated from the image data and finally we present the neural network topology we used. In section 3 we present the various results obtained. We show how the system performs for known and unknown users, as well as under changing lighting conditions. Finally, we evaluate the system in a human-robot interaction scenario.

1.1 Related Work

Generally, we distinguish two different approaches for head pose estimation:

- Feature-based approaches
- View-based approaches

Feature-based techniques try to find facial feature points in an image from which it is possible to calculate the actual head orientation. These features can be obvious facial characteristics like eyes, nose, mouth etc. View-based techniques on the other side, try to analyze the entire head image in order to decide in which direction a person's head is oriented.

Generally, feature-based methods have the limitation that the same points must be visible over the entire image sequence, thus limiting the range of head motions they can track [1]. View-based methods do not suffer from this limitation.

The exploitation of the depth information can render feature-based and view-based head pose estimation methods more accurate compared to conventional 2D approaches. We quickly present a couple of existing head pose estimation techniques, which are using depth information to improve estimation results.

Matsumoto and Zelinsky [2] proposed a template-matching technique for feature-based head pose estimation. They store six small image templates of eye and mouth corners. In each image frame they scan for the position where the templates fit best. Subsequently, the 3D position of these facial features are computed. By determining the rotation matrix M which maps these six points to a pre-defined head model, the head pose is obtained.

Harville et al. [5] used the optical flow in an image sequence to determine the relative head movement from one frame to the next. They use the brightness change constraint equation (BCCE) to model the motion in the image. Moreover they added a depth change constraint equation to incorporte the stereo information. Morency et al. [6] improved this technique by storing a couple of key frames to reduce drift.

Srinivasan and Boyer [3] proposed a head pose estimation technique using view-based eigenspaces. Morency et al. [4] extended this idea to 3D view-based eigenspaces, where they use additional depth information. They use a Kalman filter to calculate the pose change from one frame to the next. However, they reduce drift by comparing the images to a number of key frames. These key frames are created automatically from a single view of the person.

In our previous work [17] we estimated the head orientation with neural networks. As input patterns we used normalized gray value images which were scaled down to 20x30 pixels. To improve performance we added the image's horizontal and vertical edges to the input patterns. In this work we present an extension of this technique. We provide the neural networks with additional depth information and show that both accuracy and robustness of the system improve considerably.

2 Head Pose Estimation

The head pose estimation process consists of three preprocessing steps and the final estimation with neural networks. Figure 1 shows a quick overview.

2.1 Preprocessing

In the preprocessing phase first the depth information is calculated from the images of the stereo camera. We use a Videre Design MEGA-D stereo camera with its associated SVS library [15]. The SVS library is optimized for real-time processing and can do a 3D reconstruction of a scene

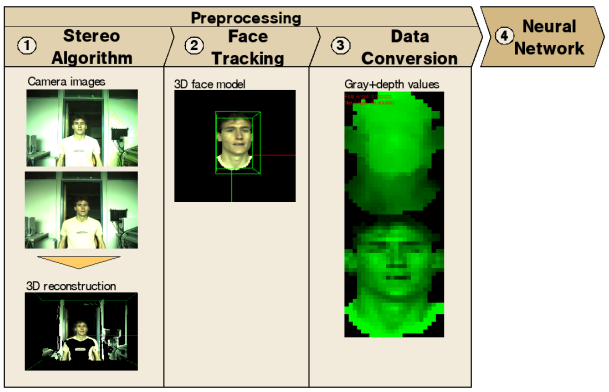

Figure 1: The components of the head pose tracker

at about 15 frames per second on a standard PC (resolution 320x240).

2.1.1 Head Detection and Extraction

Obviously, robust head detection is crucial for head pose estimation. In our system, we implemented a variation of the color-based face detection technique presented in [19].

First, a color-model for the current lighting conditions has to be constructed. This done by building a color histogram H_+ of a skin color region and a color histogram H_- of the rest of the image . The proportion of H_+ to H_- gives us an indication of the probability of a pixel to be skin-colored.

To avoid a manual initialization in the beginning, we have to find the skin color region for H_+ automatically. This is done by using another face detection algorithm from Viola and Jones [7]. This algorithm can only detect frontal faces and is significantly slower than the above color-based technique.

Once the color-model has been constructed and the pixels are classified by skin color probability, we form skin color blobs by morphological filtering. From the resulting skin color blobs, we select the one, which seems most likely to be a head.

This is accomplished by computing the real world dimensions of the color blob. Since, we know the distance of the object from the stereo reconstruction, we can calculate them with the intercept theorems. Subsequently, we can compare the dimensions of the color blob with a pre-defined head model, whose dimensions are described by a number of Gaussians.

The advantage of this method is, on the one hand, that face detection is very fast. On the other hand, the color-based face detection algorithm provides us not only with an image box which contains the face, but with the exact face mask. Already the contour of the face mask can give the neural networks an indication in which direction a head is oriented.

2.1.2 Data Conversion

After the head has been detected and extracted from the image, we have to convert the data to neural network input patterns.

Since the neural network input layer has a fixed size, firstly, we scale the extracted head down to a fixed number of pixels. We use simple decimation of the pixels for the scaling operation.

Since we do not want the neural networks to learn variations in the input patterns that do not reflect changes in the head pose, we perform a couple of normalization operations. First, we convert the color of the pixels to gray values and normalize them to values from 0 to 1. Subsequently we equalize the histogram of the resulting pattern. The equalization is accomplished by a variation of the histogram matching technique of Heeger and Bergen [9].

Histogram equalization should level some of the illumination changes, which might occur in realistic environments. However, previous results have shown that despite histogram equalization the performance of the pose estimation degrades to a great amount when the neural networks are used in illumination conditions that are different to the conditions during the training of the networks (see [18]). Consequently, our previous head pose estimation system had to be adapted to new lighting condition by using some adaptation data.

Disparity images should be much less affected by changing illumination than gray value images. By using disparity (depth) images to estimate head pose we can therefore expect an improved robustness against changing illumination. Consequently we incorporatedthe depth information in the neural network input patterns. Depth information is also normalized to values from 0 to 1 (tip of nose). In figure 1 you can see an exampleinput pattern of the system. The upper part displays the depth information, the lower the normalized gray values.

2.2 Neural Network Topology

For the neural network topology, we chose a three layer feed-forward network. Three layer feed-forward networks have the advantage that they are quite simple, but are still able to approximate any decision boundary in the high dimensional input space (see [8]).

The number of units in the input layer is crucial for a neural network. The more input units there are, the more training data has to be provided. In our system the input layer of the network consisted of 768 units if no depth information is used. Otherwise it consisted of 1536 units. The hidden layer contained 60 to 80 units.

The best results were obtained by training separate nets for pan, tilt and roll angles. Each of these networks contained a single output unit. The output of the networks is therefore directly interpreted as rotation angle normalized to values between 0 and 1.

The layers of the network were fully connected and the training was performed with the standard back-propagation algorithm.

Unlike other head pose estimation techniques, we do not estimate the relative head rotation from one frame to another with this approach. We directly compute the orientation from a single image frame. That is why the estimation errors aren't accumulated over an image sequence. Moreover, there is no need for the tracking system to know the user's initial head orientation.

An advantage of the above network topology is, that we do not divide the estimation in angle ranges or classes. Consequently, the real head orientations can be approximated very precisely.

Once neural networks are trained, they are extremely fast in computation. The activation levels of the input patterns only have to be propagated through the three layers of the network. As we will see in the following chapter, neural networks are accurate as well. Hence, they are well suited for a real-time head pose estimation technique.

3 Experimental Results

For evaluating the developed head pose estimation technique, we have done two data collections.

The "Portrait View" data collection has been recorded in a relatively restricted environment. People were sitting in about two meter distance in front of the stereo camera. The people's head movement wasn't restricted in any way. They were free to move their heads in pan, tilt and roll direction. The recorded rotation angles ranged from $-90°$ to $90°$

Since one of our main goals was to evaluate the head pose estimation accuracy under changing lighting conditions. The data was recorded under two different illuminations. One scenario therefore consisted of a room illuminated by day light, in the other scenario the room was illuminated artificially with neon lamps. In order to obtain an even stronger effect of the illumination change, we tried to place an additional lamp next to the person. This was done to intensify the shadows in the face. Shadows shouldn't have an effect on the stereo reconstruction, but certainly affect the angle estimation with a conventional intensity image-based technique.

Figure 2 shows some sample pictures from the data collection.

In total we recorded image sequences of 10 persons looking around freely. The image sequences consisted of approximately 500 pictures and were recorded under both of the lighting conditions described above. The real head orientations were tracked with a magnetic sensor (Flock of Birds). Image resolution was 640x480 pixels.

For the evaluation we mainly focused on the pan angle. Pan direction is, on the one hand, the direction where the most movement occurs, on the other hand, the pan angle seems to be the most useful angle for identifying the object a person is focusing on. For completeness, however, we also provide the results obtained for the tilt angle.

In order to have results, which are comparable to our old system [18] and to see the difference of performance with and without depth information, we tested our method with histogram normalized gray value images (24x32 pixels), depth images (24x32 pixels) and a combination of gray value and depth images (2x24x32 pixels).

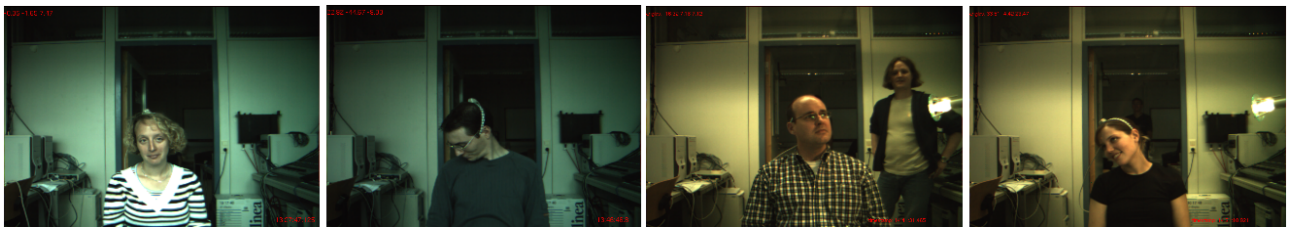

Figure 2: Sample images from the "Portrait View" data collection. The first two images are recorded with daylight, the others under artificial illumination

3.1 Multi-User And New User Performance

First we evaluated the system's performance on multiple users (multi-user test). Here, the neural network was trained and evaluated on all persons of the data set.

Column 2 of table 1 shows the results, we obtained for this test.

mean error	multi-user	new user
gray values	4.2 / 2.9	9.6 / 8.8
depth info	5.1 / 3.8	11.0 / 7.6
depth + gray	**3.2** / **2.6**	7.5 / 6.7

Table 1: Mean error obtained for the multi-user and new user tests in the "Portrait View" scenario (pan/tilt angles)

As you can see, patterns consisting of gray values achieve rather good results. However, when we combine them with additional depth information, we are able to improve the mean error for the pan angle by about 24%. Patterns containing only depth information are less accurate.

However, for practical applications, we do not want the system to depend on the user. This would imply to retrain the network for every new user. Since neural network training is computational expensive we want to avoid retraining.

Column 3 of table 1 shows the results, we obtained when the network was trained on 9 persons and evaluated on the remaining one.

Again, we observe that patterns consisting of a combination of gray value and depth information achieve the best result. The relative improvement compared to gray value patterns is in this case 25% for the pan angle.

The absolute values, however, are considerably worse than the results for multi-user test. This is due to a number of circumstances. Firstly, heads differ in appearance and aspect ratio. For example, some people's heads are rather longish, whereas others have heads which are quite broad. Another issue is the performance of the face tracking technique. Since the heads in our system were automatically extracted, sometimes hair is considered to partially belong to the face and sometimes not. Especially for people with long hair, the estimation was considerably worse. However, with a mean error of $7.2°$ the estimation still is rather good for many applications.

In order to illustrate this further, we analysed the number of angle estimates for horizontal orientation (pan) with an error smaller than $10°$ and $20°$ respectively. In this context, we call the percentage of angles satisfying this criterion *direction recognition rate*.

Table 2 compares the recognition rates for the multi-user and new user case.

recognition rate	error $< 10°$	error $< 20°$
multi-user	94.3%	99.7%
new user	75.2%	95.1%

Table 2: Direction recognition rates for head pan obtained for the multi-user and new user tests in the "Portrait View" scenario

3.2 Changed Lighting Conditions

Changing lighting conditions are one of the main problems of image-based techniques and particularly neural networks. In our previous work we observed a significant degredation of pose estimation accuracy, when the neural networks were used in new lighting conditions. Therefore, the networks had to be adapted to new illumination conditions by collecting some additional training data in the new environment [18]. Depth information should be useful to improve the robustness against such illumination changes.

In order to evaluate the system's performance under changing illumiation conditions, we now trained a neural network on all user sequences which have been recorded in the room illuminated with day light. Then, we tested the networks on the data recorded in the room which was illuminated by neon lamps and strong side illumination.

Figure 3 shows the obtained results under these conditions for the pan angle.

The mean deviation from the reference angle with normalized gray value patterns increases to $13.9°$. The combination of gray value and depth information leads to a mean deviation of $10.6°$, whereas under these circumstances depth information alone achieves with $9.6°$ mean deviation the best result.

For the direction recognition rates we achieved values of up to 60% and 87.6%.

In our old system [17] we tried to improve the head pose estimation with edge information obtained from the image. However, when lighting conditions changed, the performance of this system still decreased significantly more than in the system presented here. There we achieved a mean deviation of only $13.8°$.

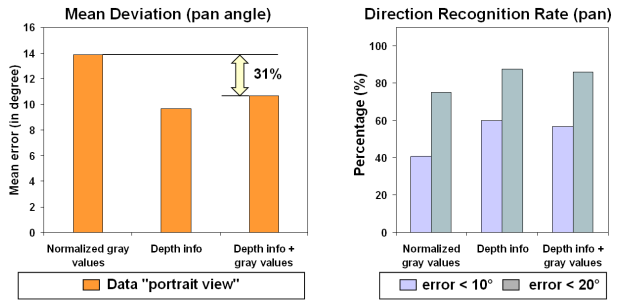

Figure 3: Mean deviation from the reference angle and direction recognition rates with changed lighting conditions

We conclude that depth information is well suited for head pose estimation under changing lighting conditions. To have a versatile method working well under all conditions, we propose nevertheless to combine depth and gray value information for head pose estimation. The conventional intensity image-based approach is still 31% worse with this configuration.

3.3 Human Robot Interaction

In the second data collection, we tried to build a realistic setup for a human-robot interaction application. Now the users were further away from the camera. They were not only free to turn their head in any direction, but also to move around in the room.

A total of six users have been recorded under these conditions. The data sequences consist of about 1000 images per person with a resolution of 640x480 pixels.

The goal of the experiments was to compare the system's performance to the results under the more restricted conditions in the "Portrait View" data collection and to obtain a pose estimation system that would work in a realistic human-robot interaction scenario.

Table 3 shows the results for mulit-user and new user tests in this scenario.

mean error	multi-user	new user
gray values	4.6 / 2.4	15.5 / 6.3
depth info	8.0 / 3.3	11.0 / 5.7
depth + gray	**4.3** / **2.1**	9.7 / 5.6

Table 3: Mean error obtained for the multi-user and new user tests in the human-robot interaction scenario (pan/tilt angles)

With a mean deviation of only $4.3°$ for known users and $9.7°$ for new users the result is still very good. However, the depth information hasn't been able to improve the result as much as for the "Portrait View" data collection. This is due to the fact, that the stereo information becomes worse the farther away an object is.

For the direction recognition rates we achieved values of 91.5% and 98.6% for known users as well as 63.1% and 90.4% for unknown users. These values show that this method is applicable in practice.

recognition rate	error $< 10°$	error $< 20°$
multi-user	91.4%	98.6%
new user	63.1%	90.4%

Table 4: Direction recognition rates for head pan obtained for the multi-user and new user tests in the human-robot interaction scenario

3.4 Kalman Filter

In order to further improve the achieved results, we implemented a Kalman filter. Figure 5 shows an example curve of the real, estimated and filtered angle in an image sequence for a new user in the "Robot Scenario".

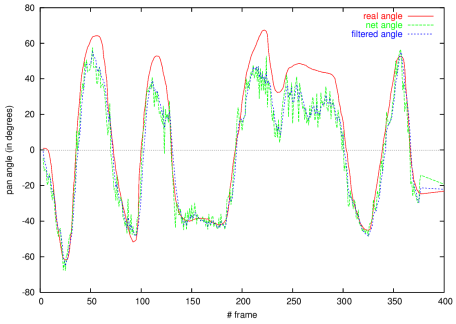

Figure 5: Real, estimated and Kalman-filtered pan angles in an image sequence

It can be seen, that both the estimated and filtered angle are pretty close to the real rotation angle. The Kalman filtered curve is a little smoother and reduces the overall error from $9.7°$ to $9.1°$ for new users in the "Robot Scenario".

3.5 Integration

To prove the practicability of the technique, we integrated it into a fully automatic, real-time head pose estimation system for human-robot interaction. A user may walk into the scene, where his face will be detected automatically to update the color-model. Subsequently, his head is extracted from each reconstructed stereo image in real-time. The head image is converted into an input pattern and a trained neural network, which is loaded into the system, outputs the current head orientation.

At a resolution of $320x240$ pixels, we achieve a frame rate of more than 10 fps on a standard PC. The main part of the computation time is consumed for the stereo reconstruction.

Furthermore, we integrated the head pose estimation technique into a pointing gesture recognition system presented in [19]. When pointing at an object, people also tend to look in the same direction. Hence, the accuracy of the gesture recognition system should improve, when the head

Figure 4: Sample images from the ”Robot Scenario” data collection

orientations are known. The results of this work have been submitted to FG'2004 separately.

4 Summary and Conclusions

In this paper we proposed a head pose estimation technique, for human-robot interaction. The technique is capable of real-time processing and does not need any manual initialization. Our evaluation showed that head orientations from -90° to $+90^\circ$ can be tracked accurately even in non-restricted environments and when users are further away from the camera. Furthermore we showed that depth information is not only capable of improving the system's accuracy for known and unknown users, but also significantly improves robustness when lighting conditions change.

Acknowledgements

This research is partially supported by the German Research Foundation (DFG) as part of the Sonderforschungsbereich 588 “Humanoide Roboter”.

References

[1] R. Yang and Z. Zhang, “Model-based Head Pose Tracking With Stereovision”, *FG 2002*, pp. 255-260, Washington DC, 2002

[2] Y. Matsumoto, A. Zelinsky, “An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement”, *FG 2000*, pp.499-505, 2000

[3] S. Srinivasan and K. L. Boyer, “Head Pose Estimation Using View Based Eigenspaces”, *ICPR 2002*, Quebec, 2002

[4] Morency, Sundberg, Darrel, “Pose Estimation using 3D View-Based Eigenspaces”, *IEEE Intl. Workshop on Analysis and Modeling of Faces and Gestures*, pp. 45-52, Nice, 2003

[5] M. Harville, A. Rahimi, T. Darell, G. Gordon, J. Woodfill, “3D Pose Tracking with Linear Depth and Brightness Constraints”, *ICCV'99*, Corfu, Greece, 1999

[6] Morency, Rahimi, Checka and Darrell, “Fast stereobased head tracking for interactive environment”, *FG 2002*, Washington DC, 2002

[7] P. Viola and M. Jones, “Robust real-time object detection”, *Technical Report 2001/01*, Compaq CRL, February 2001.

[8] Christopher M. Bishop, “Neural Networks for Pattern Recognition”, *Oxford University Press*, 2000

[9] D. Heeger and J. Bergen, “Pyramid-based texture analysis/synthesis”, *SIGGRAPH 1995*, pages 229-238, 1995

[10] D. Perzanowski et al., “Building a multimodal human-robot interface”, *IEEE Intelligent Systems*, pages 16-21,2001

[11] A. Agah, “Human interactions with intelligent systems: research taxonomy”, *Computers and Electrical Engineering*, pages 71-107, 2001

[12] A. Koku et al., “Towards socially acceptable robots”, *Intl. Conf. on Systems, Man and Cybernetics*, pages 894-899, 2000

[13] , B. Adams et al., “Humanoid robots: a new kind of tool”, *IEEE Intelligent Systems*, pages 25-31, 2000

[14] , Y. Matsusaka et al., “Multi-person conversation via multi-modal interface - A robot who communicates with multi-user”, *Eurospeech'99*,pages 1723-1726, 1999

[15] K. Konolige, “Small Vision System: Hardware and Implementation”, *IEEE Conference on Computer Eighth International Symposium on Robotics Research*, Hayama, Japan, 1997

[16] K. Nickel and R. Stiefelhagen, “Detection and Tracking of 3D-Pointing Gestures for Human-Robot-Interaction”, *Humanoids 2003*, Karlsruhe, Germany, 2003

[17] Rainer Stiefelhagen, Jie Yang, Alex Waibel, “Simultaneous Tracking of Head Poses in a Panoramic View”, *ICPR 2000*, Barcelona, Spain, 2000.

[18] Rainer Stiefelhagen, “Tracking Focus of Attention in Meetings”, *IEEE International Conference on Multimodal Interfaces*, pp. 273-280, Pittsburgh, USA, 2002

[19] K. Nickel and R. Stiefelhagen, “Pointing Gesture Recognition based on 3Dtracking of Face, Hands and Head Orientation”, *International Conference on Multimodal Interfaces*, Vancouver, Canada, 2003

[20] R. Stiefelhagen, J. Yang, A. Waibel, “Tracking Focus of Attention for Human-Robot Communication”, *Humanoids 2001*, Tokyo, Japan, 2001

[21] Kai Nickel, Edgar Seemann, Rainer Stiefelhagen, “3D-Tracking of Head and Hands for Pointing Gesture Recognition in a Human-Robot Interaction Scenario”, *FG 2004*, Seoul, Korea, 2004