Control and recognition on a humanoid head with cameras having different field of view

Aleš Ude
Jožef Stefan Institute, ABR
Jamova 39, Ljubljana, Slovenia
ales.ude@ijs.si

Tamim Asfour
University of Karlsruhe, ITEC
Haid-und-Neu-Strasse 7, Karlsruhe, Germany
asfour@ira.uka.de

Abstract

In this paper we study object recognition on a humanoid robotic head. The head is equipped with a stereo vision system with two cameras in each eye, where the cameras have lenses with different view angles. Such a system models the foveated structure of a human eye. To facilitate the pursuit of moving objects, we provide mathematical analysis that enables the robot to guide the narrow-view cameras toward the object of interest based on information extracted from the wider views. Images acquired by narrow-view cameras, which produce object images at higher resolutions, are used for recognition. The proposed recognition approach is view-based and is built around a classifier using non-linear multi-class support vector machines with a special kernel function. We show experimentally that the increased resolution leads to higher recognition rates.

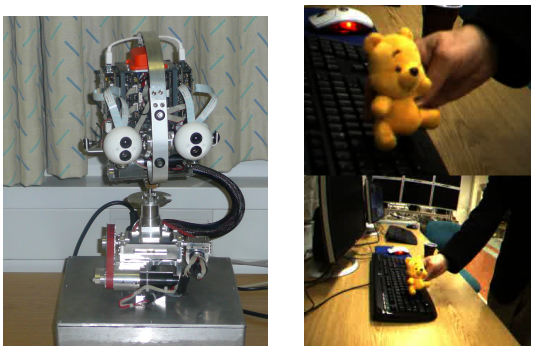

Figure 1. An example humanoid head (left). The narrow-angle cameras are positioned above the wide-angle ones. On the right are the images simultaneously taken from the wide- and narrow-angle view.

1 Introduction

Designers of a number of humanoid robots attempted to replicate human oculomotor system. For the optical part, this means that the optics should model the foveated structure of the human eye and allow *simultaneuos* processing of images of varying resolution. For the motor part, this means that the head must have sufficent mobility to perform typical eye movements such as smooth pursuit and saccades. Such an arrangement is useful because, firstly, it enables the robot to monitor and explore its surroundings in wide-angle views that contain most of the environment at low resolution, thereby increasing the efficiency of the search process. Secondly, it makes it possible to simultaneously extract additional information – once the area of interest is determined – from narrow-angle camera images that contain more detail. This kind of system is especially useful for object recognition on a humanoid robot. General object recognition is difficult because it requires the robot to detect objects in dynamic environments and to control the eye gaze to get the objects into the fovea and to keep them there. These tasks can be accomplished using information from wide-angle views, which enables the robot to determine the identity of the object by processing narrow-angle views.

There are various ways to construct humanoid vision system in hardware. The approach we followed is is to use two cameras in each eye equipped with lenses with different focal lengths [1, 4, 5]. This has the advantage of allowing us to use small-form cameras for the construction of the head.

2 Wide- and Narrow-Angle Views

The humanoid head of Fig. 1 has narrow-angle cameras rigidly connected to the wide-angle cameras and

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

placed above them with roughly aligned optical axes. In the following we show that objects can be placed in the central field of view of narrow-angle cameras by bringing them to a certain position in the wide views. This position is displaced from the center of wide-angle camera images. The necessary displacement depends on the distance of the object from the cameras.

For theoretical analysis, we model both cameras by a standard pinhole camera model. The relationship between a 3-D point $\boldsymbol{M} = [X, Y, Z]^T$ and its projection $\boldsymbol{m} = [x, y]^T$ is given by

$$s\tilde{\boldsymbol{m}} = \boldsymbol{A}\tilde{\boldsymbol{M}},\ \boldsymbol{A} = \begin{bmatrix} \alpha & \gamma & x_0 \\ 0 & \beta & y_0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (1)$$

where $\tilde{\boldsymbol{M}} = [\boldsymbol{M}^T, 1]^T$, $\tilde{\boldsymbol{m}} = [\boldsymbol{m}^T, 1]^T$ are both points in homogeneous coordinates, s is an arbitrary scale factor, and $(\alpha, \beta, \gamma, x_0, y_0)$ are the intrinsic parameters of the camera. The world coordinate system is assumed to coincide with the camera coordinate system. In the following we assume without loss of generality that the origin of the image coordinate system coincides with the principal point (x_0, y_0), thus $x_0 = y_0 = 0$.

Let $\hat{\boldsymbol{t}}$ be the position of the origin of the wide-angle camera coordinate system expressed in the narrow-angle camera coordinate system and let $\hat{\boldsymbol{R}}$ be the rotation matrix that rotates the basis vectors of the wide-angle camera coordinate system into the basis vectors of the narrow-angle camera coordinate system. We denote by $\boldsymbol{M}_n$ and $\boldsymbol{M}_w$ the position of a 3-D point expressed in the narrow- and wide-angle camera system, respectively. We then have

$$\boldsymbol{M}_w = \hat{\boldsymbol{R}}(\boldsymbol{M}_n - \hat{\boldsymbol{t}}). \quad (2)$$

The projections of a 3-D point $\boldsymbol{M}_n = (X, Y, Z)$ onto the planes of both cameras are given by

$$x_n = \frac{\alpha_n X + \gamma_n Y}{Z}, \quad (3)$$

$$y_n = \frac{\beta_n Y}{Z}, \quad (4)$$

and

$$x_w = \frac{\alpha_w \boldsymbol{r}_1 \cdot (\boldsymbol{M}_n - \hat{\boldsymbol{t}}) + \gamma_w \boldsymbol{r}_2 \cdot (\boldsymbol{M}_n - \hat{\boldsymbol{t}})}{\boldsymbol{r}_3 \cdot (\boldsymbol{M}_n - \hat{\boldsymbol{t}})}, \quad (5)$$

$$y_w = \frac{\beta_w \boldsymbol{r}_2 \cdot (\boldsymbol{M}_n - \hat{\boldsymbol{t}})}{\boldsymbol{r}_3 \cdot (\boldsymbol{M}_n - \hat{\boldsymbol{t}})}, \quad (6)$$

where $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, and $\boldsymbol{r}_3$ are the rows of the rotation matrix $\hat{\boldsymbol{R}} = \begin{bmatrix} \boldsymbol{r}_1^T & \boldsymbol{r}_2^T & \boldsymbol{r}_3^T \end{bmatrix}^T$. $\boldsymbol{M}_n$ projects onto the principal point in the narrow-angle camera if $x_n = y_n = 0$. Assuming that the point is in front of the camera, hence $Z > 0$, we obtain from Eq. (3) and (4) that $X = Y = 0$, which means that the point must lie on the optical axis of the narrow-angle camera. Inserting this into Eq. (5) and (6), we obtain the following expression for the ideal position $(\hat{x}_w, \hat{y}_w)$ in the wide-angle camera image that results in the projection onto the principal point in the narrow-angle camera image

$$\hat{x}_w = \frac{\alpha_w \boldsymbol{r}_1 \cdot \hat{\boldsymbol{t}} + \gamma_w \boldsymbol{r}_2 \cdot \hat{\boldsymbol{t}} - (\alpha_w r_{13} + \gamma_w r_{23})Z}{\boldsymbol{r}_3 \cdot \hat{\boldsymbol{t}} - r_{33}Z}, \quad (7)$$

$$\hat{y}_w = \frac{\beta_w \boldsymbol{r}_2 \cdot \hat{\boldsymbol{t}} - \beta_w r_{23} Z}{\boldsymbol{r}_3 \cdot \hat{\boldsymbol{t}} - r_{33}Z}, \quad (8)$$

where $\begin{bmatrix} r_{13} & r_{23} & r_{33} \end{bmatrix}^T$ is the third column of $\hat{\boldsymbol{R}}$. Note that the ideal position in the periphery is independent from the intrinsic parameters of the foveal camera. It depends, however, on the distance of the point of interest from the cameras.

Utilizing these formulas we can turn the eye gaze towards the object and keep the object in the center of narrow-angle cameras based on information from wide-angle views. This is important because it is difficult to move the cameras quick enough to keep the object in the center of narrow-angle views. For this reason the object can easily be lost from narrow-angle views. Therefore it is advantageous to control the cameras using information from wide-angle views.

3 Learning Object Representations

We developed an object tracking system [6] that allows the robot to find objects of interest and locate them in the images. Using the formulas described in Section 2 and stereo vision, the robot can apply the results of the tracking process to center the object of interest in the narrow-angle view, where the object image has relatively high resolution. Since our tracker can estimate both the location and scale of the object in the image, we can warp, i. e. translate, rotate and scale along the principal axes, the object images to a window of constant size.

Our goal is to learn a view-based representation for all available objects. To achieve this, it is necessary to show the objects to the humanoid from all relevant viewing directions. In computer vision this is normally achieved by accurate turntables that enable the collection of images from regularly distributed viewpoints. However, this solution is not practical for humanoid robotics, where on-line interaction is often paramount. We therefore explored whether it is possible to reliably learn models from images collected while a human teacher randomly moves the object in front of the robot. In this case the training process is started by a

teacher who moves the object to be learnt in front of the robot. Snapshots from various viewpoints are collected and processed. Warping the snapshots onto a window of constant size ensures invariance against scaling and planar rotations.

To ensure maximum classification performance, the data is further processed before training a general classifier. Most modern view-based approaches characterize the views by ensembles of local features. We use complex Gabor kernels to identify local structure in the images. A Gabor jet at pixel $\boldsymbol{x}$ is defined as a set of complex coefficients $\{J_j^{\boldsymbol{x}}\}$ obtained by convolving the image with a number of Gabor kernels at this pixel. The kernels are normally selected so that they sample a number of different wavelengths k_ν and orientations ϕ_μ. Wiskott et al. [7] proposed to use $k_\nu = 2^{-\frac{\nu+2}{2}}$, $\nu = 0, \ldots, 4$, and $\phi_\mu = \mu\frac{\pi}{8},\ \mu = 0, \ldots, 7$, but this depends both on the size of the incoming images and the image structure. They showed that the similarity between the jets can be measured by

$$\mathrm{S}\left(\{J_i^{\boldsymbol{x}}\}, \{J_i^{\boldsymbol{y}}\}\right) = \frac{\boldsymbol{a}_{\boldsymbol{x}}^T * \boldsymbol{a}_{\boldsymbol{y}}}{\|\boldsymbol{a}_{\boldsymbol{x}}\|\|\boldsymbol{a}_{\boldsymbol{y}}\|}, \tag{9}$$

where $\boldsymbol{a}_{\boldsymbol{x}} = [|J_1^{\boldsymbol{x}}|, \ldots, |J_s^{\boldsymbol{x}}|]^T$ and s is the number of complex Gabor kernels. This is based on the fact that the magnitudes of complex coefficients vary slowly with the position of the jet in the image.

Our system builds feature vectors by sampling Gabor jets on a regular grid of pixels $\boldsymbol{X}_G$. At each grid point we calculate the Gabor jet and add it to the feature vector. The grid points need to be parsed in the same order in every image. The grid size used in our experiments was 6×6, the warped image size was 160×120 with pixels outside the enclosing ellipse excluded, and the dimension of each Gabor jet was 40, which resulted in feature vectors of dimension 16080. These feature vectors were supplied to the SVM for training.

4 Nonlinear Multi-Class SVMs

Utilizing the similarity measure (9), we developed a classifier for object recognition based on nonlinear multi-class support vector machines. Nonlinear multi-class support vector machines (SVMs) [2] use the following decision function

$$\boldsymbol{H}(\boldsymbol{x}) = \arg\max_{r \in \boldsymbol{\Omega}} \left\{ \sum_{i=1}^{m} \tau_{i,r} \mathrm{K}(\boldsymbol{x}_i, \boldsymbol{x}) + b_r \right\}. \tag{10}$$

Here $\boldsymbol{x}$ is the input feature vector to be classified (in our case a collection of Gabor jets), $\boldsymbol{x}_i$ are the feature vectors supplied to the SVM training, $\tau_{i,r},\ b_r$ are the values estimated by SVM training, and $\Omega = \{1, \ldots, N\}$ are the class identities (objects in our case). The feature vectors $\boldsymbol{x}_i$ with $\tau_{i,r} \neq 0$ are called the support vectors. The SVM training consists of solving a quadratic optimization problem whose convergence is guaranteed for all kernel functions K that fulfill the Mercer's theorem.

The similarity measure for Gabor jets (9) provides a good motivation for the design of a kernel function for the classification of feature vectors consisting of Gabor jets. Let $\boldsymbol{X}_G$ be the set of all grid points within two normalized images on which Gabor jets are calculated and let $J_{\boldsymbol{X}_G}$ and $L_{\boldsymbol{X}_G}$ be the Gabor jets calculated in two different images, but on the same grid points. A suitable kernel function can be defined as follows

$$\mathrm{K}_G(J_{\boldsymbol{X}_G}, L_{\boldsymbol{X}_G}) = \exp\left(-\rho \tfrac{1}{M} \textstyle\sum_{\boldsymbol{x} \in \boldsymbol{X}_G} \left(1 - \frac{\boldsymbol{a}_{\boldsymbol{x}}^T * \boldsymbol{b}_{\boldsymbol{x}}^T}{\|\boldsymbol{a}_{\boldsymbol{x}}\|\|\boldsymbol{b}_{\boldsymbol{x}}\|}\right)\right), \tag{11}$$

where M is the number of grid points in $\boldsymbol{X}_G$. This function satisfies the Mercer's condition [2] and can thus be used for support vector learning. Parameter ρ needs to be supplied experimentally.

5 Experimental Results

We used a set of ten objects to test the performance of the developed recognition system on a humanoid robot. For each object we recorded two or more movies using a video stream coming from the narrow-angle cameras, which were controlled by information acquired from wide-angle views. In each of the recording sessions the teacher attempted to show one of the objects to the robot from all relevant viewing directions. One movie per object was used to construct the SVM classifier, while one of the other movies was used to test the classifiers. Each movie was one minute long and we used at most 4 images per second for training. Since slightly more than first ten seconds of the movies were needed to initialize the tracker, we had at most 208 training images per object. For testing we used 10 images per second, which resulted in 487 test images per object. All the percentages presented here were calculated using the classification results obtained from 4870 test images. Gabor jets were calculated as proposed by Wiskott et al. [7] and the grid size was 6 pixels in both directions. The filters were scaled appropriately when using lower resolution images. To show the usefulness of foveated vision for recognition, we tested the performance of the system on images of varying resolution. We also compared the developed SVM-based classifier with the nearest neighbor classifier (NNC) that uses the similarity measure (9) – summed over all grid points – to deterrmine the class of the nearest neighbor by comparing Gabor jets directly.

Table 1. Correct classification rate (image resolution 120×160 pixels)

Training views per object	SVM	NNC
208	97.6 %	95.9 %
104	96.7 %	93.7 %
52	95.1 %	91.5 %
26	91.9 %	86.7 %

Table 2. Correct classification rate (image resolution 60×80 pixels)

Training views per object	SVM	NNC
208	94.2 %	89.3 %
104	92.4 %	87.3 %
52	90.7 %	84.4 %
26	86.7 %	79.2 %

Table 3. Correct classification rate (image resolution 30×40 pixels)

Training views per object	SVM	NNC
208	91.0 %	84.7 %
104	87.2 %	81.5 %
52	82.4 %	77.8 %
26	77.1 %	72.1 %

Results in Tables 1 - 3 prove that foveation is very useful for recognition. The classification results clearly become worse with the decreasing resolution. Our results also show that we can collect enough training data even without using accurate turntables to systematically collect the views. As expected the recognition rate decreases with the number of images, but we can conclude that collecting the training views statistically is sufficient to build models for 3-D object recognition.

The presented results cannot be directly compared to the results on standard databases for benchmarking object recognition algorithms because here the training sets are much less complete. Some of the classification errors are caused by the lack of training data rather than by a deficient classification approach. Unlike many approaches from the computer vision literature that avoid the problem of finding objects, we tested the system on images obtained through a realistic object tracking and segmentation procedure. Only such data is relevant for foveated object recognition because without some kind of segmentation it is not possible to direct the fovea towards the objects of interest.

6 Conclusions

Our experiments demonstrate that by exploiting the properties of a humanoid vision we can construct an effective object recognition system. Wide-angle views are necessary to search for objects, direct the gaze towards them and keep them in the center of narrow-angle views. Narrow-angle views provide object images at a higher resolution, which significantly improves the recognition rate. Having both views at the same time is essential. Most of previous approaches that employed support vector machines for object recognition used binary SVMs combined with decision trees [3]. Our system makes use of nonlinear multi-class SVMs to solve the multi-class recognition problem. By normalizing the views with respect to scale and planar rotations based on the results of the tracker, we were able to reduce the amount of data needed to train the SVMs. Object representations can be learnt just by collecting the data statistically while the demonstrator attempts to show the objects from all relevant viewing directions. Experimental results show high recognition rates in realistic test environments.

Acknowledgment:The work described in this paper was partially conducted within the EU Cognitive Systems project PACO-PLUS (FP6-2004-IST-4-027657) funded by the European Commission.

References

[1] C. G. Atkeson, J. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaal, T. Shibata, G. Tevatia, A. Ude, S. Vijayakumar, and M. Kawato. Using humanoid robots to study human behavior. *IEEE Intelligent Systems*, 15(4):46–56, July/August 2000.

[2] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Machine Learning Research*, 2:265–292, 2001.

[3] G. Guo, S. Z. Li, and K. L. Chan. Support vector machines for face recognition. *Image and Vision Computing*, 19(9-10):631–638, 2001.

[4] H. Kozima and H. Yano. A robot that learns to communicate with human caregivers. In *Proc. Int. Workshop on Epigenetic Robotics*, Lund, Sweden, 2001.

[5] B. Scassellati. A binocular, foveated active vision system. Technical Report A.I. Memo No. 1628, MIT, Artificial Intelligence Laboratory, 1999.

[6] A. Ude and C. G. Atkeson. Probabilistic detection and tracking at high frame rates using affine warping. In *Proc. 16th Int. Conf. Pattern Recognition, Vol. II*, pages 6–9, Quebec City, Canada, 2002.

[7] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Machine Intell.*, 19(7):775–779, 1997.