Gabor Filters for Object Localization and Robot Grasping

Jörg Walter · Bert Arnrich
Department of Computer Science · University of Bielefeld
D-33501 Bielefeld · Email: walter@techfak.uni-bielefeld.de

Abstract

We present a system for learning the 3 DOF fine-positioning task of a robot manipulator (Puma 260) using a gripper mounted camera. Small lateral gripper-target misalignments are corrected in one step. Larger ones employ a previous coarse adjustment move in order to bound the parallax effects of the close camera focus. We build object-specialized, neural network-based pose estimators with a rather small set of Gabor filters. Gabor filters perform a spatially localized frequency analysis and resemble the spatial response profile of receptive fields found in visual cortex neurons. The system demonstrates efficiency w.r.t. speed and accuracy, as well as robustness against changing illumination and object conditions.

1. Introduction

Neural networks were evolved by nature to enable perception and action. Therefore, artificial neural networks seem as an appropriate choice for learning one of the most demanding sensor-based manipulation skills – grasping. One of the main obstacles for industrial applications is the availability of robust and inexpensive sensor–action systems. As the price and size of reasonable vision systems is decreasing, camera sensors become a popular expansion to robot systems. Traditional robot vision research focuses on the use of explicit world models and their construction from raw sensor data. Scene reconstruction is undoubtedly useful but expensive and often too complex in a changing real-world environment. We think that locally operating learning schemes are the best candidates to advance intelligent robot systems. This leads to the question what are the best feature extraction approaches for feeding a learning network. Previous work [6] employed line segment Hough Transforms (e.g. [2]) and [7] used principal component analysis (PCA) and appearance-based eigenimages [5] for same task. We will briefly compare their results with our Gabor-filter-based approach.

Gabor wavelets proved advantageous for object and face recognition [4], there acknowledged for their pose invariance. Here, we examine the reverse task. Knowing the object, how well can we estimate the pose? And, can we build a robust and efficient sub-system?

System Overview: The overall aim of our system is the structure assembling demonstrated using a set of wooden pieces including nuts, screws, ledges, and cubes, see Fig. 1. This task can be divided into several smaller ones: a single piece has to be identified, the gripper is brought in a suitable pre-grasp position, the target is firmly enclosed and gets finally transfered to the desired mating/assembly position with other parts. The robot system consists of a 6 DOF manipulator with a camera attached to the parallel yaw gripper with a tilted viewing angle.

Grasping without any alignment help requires that the objects are picked with certain precision. Failures include the risk of *(i)* object–gripper collisions, *(ii)* pushing/displacing something before yaw closure, and *(iii)* bad object-in-gripper alignment (creating trouble for part mating later).

Here we discuss the critical pre-grasp phase, i.e. the 3 DOF fine-positioning of the manipulator after an initial coarse positioning has been completed. This implies that the resting object is visible inside the viewing angle of the hand camera and its type and vertical position is known. Now the system has to deal with significantly changing appearance of the target objects with respect to *(i)* the lo-

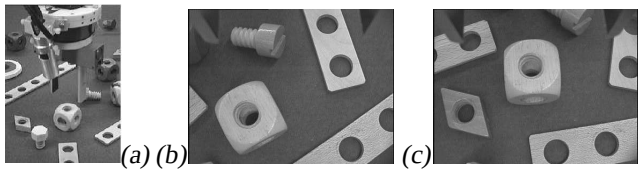


Figure 1. The end-effector over the target: *(a)* the gripper and the hand camera. A "cube" viewed by the hand camera before *(b)* – and after the fine positioning *(c)*. Note the tool tips in the upper rim of *(b,c)*.

cal lighting situation (occlusion of lamps by the robot itself, interfering humans, etc.), *(ii)* image contrast and color (several possible object colors), *(iii)* parallax effects by the camera viewing from a close and tilted position.

2. Object Representation With Gabor Filters

In order to efficiently employ a learning neural network for pose estimation we need a suitable object representation gained from the sensory input, in our case a camera image. "Suitable" means here, that the feature set is of minimal size – providing the desired accuracy and, as a consequence, more rapid learning with fewer input neurons.

Biology gave us inspiration: 1987, Jones and Palmer [3] showed by cat visual cortex experiments, that receptive fields of simple cells fit well to a profile model, previously suggested by Daugman 1980 [1]. This model describes the spatial sensitivity by a 2D extention of Gabor's work (1946, originally in the time domain). By a local formulation of the frequency content he created a "localized" Fourier analysis, here written as a complex kernel function:

$$\Psi_{\lambda\sigma\alpha}(x,y) = \exp\left(-\frac{x^2+\alpha^2y^2}{2\sigma^2}\right)\exp\left(-2\pi i\frac{x}{\lambda}\right) \quad (1)$$

Eq. 1 describes a Gaussian bell function – modulating a planar wave. The wave has the period length λ in x-direction; the elliptical Gaussian has a longitudinal width σ and σ/α transversal (aspect ratio α).

Equation 1 can be called *mother wavelet* and a complete family of self-similar *daughter* wavelets (sometimes called *jet*) can be constructed by the generating function

$$\begin{aligned}\Psi_{pqm\theta\lambda\sigma\alpha}(x,y) = &\quad 2^{-2m}\Psi_{\lambda\sigma\alpha}(x',y')\\ x' = &\quad 2^{-m}[+x\cos\theta + y\sin\theta] - p \quad (2)\\ y' = &\quad 2^{-m}[-x\sin\theta + y\cos\theta] - q.\end{aligned}$$

Here the substituted variables incorporate dilations of the wavelet in size 2^{-m}, translations in position (p,q), and rotations through the angle θ.

Each cell's receptive field can be modeled by a Gabor wavelet function, parameterized by the center (p,q), the wavelength $\lambda/2^m$ in direction θ with Gaussian elliptic envelope (with width $\sigma/2^m$ and $\sigma/2^m\alpha$) and a complex phase angle ψ (projecting a mixture of the real and imaginary part).

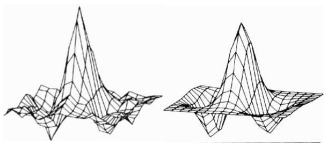


Figure 2. The 2D Gabor filter *(right)* fits simple cell spatial response profile *(left)* of receptive fields in cat striate cortex neurons [3]. See also Fig. 4.

Our system uses a collection of n those artificial neurons, all looking at the same image but each with a different receptive field. Thus the seen object gets represented by n values, i.e. the scalar product of the image by the appropriate Gabor filter mask. For a larger group of neurons, which differ only in their center position (p,q), the procedure can be speeded up by performing a convolution and implementing it as a product in the 2D-Fourier space.

3. Experimental Setup

The "Cubical" Challenge: In the following, we focus on one target object, the cubical wooden piece already shown in Fig. 1. Its size is little smaller than the open gripper and therefore calls for grasping tolerance of about 2 mm and 5-8°. The wooden "cube" is challenging for machine vision: *(i)* due to the widely rounded corners its facial surfaces are actually ring shaped; *(ii)* the three axial screw holes cause peculiar shadows. *(iii)* the contour, seen from an tilted viewing angle, resembles an egg with moving bumps when rotating the "cube". One way to avoid latter problem, is to turn the end-effector such, that for image grabbing the camera is looking vertically downwards. The price is a robot transfer delay (decrease of operation speed) and extra kinematic restrictions within the robot's workspace.

Pre-Grasp Procedure: In order to allow vivid and accurate operation we keep the gripper vertical and subdivide the fine positioning in two parts: *(i)* a fast and coarse – pure translational part and *(ii)* a rotational/translational fine part. Fig. 3 displays the simplified procedure in an UML-activity diagram. The grabbed image is preprocessed and the object's center of gravity in image coordinates (u,v) mapped by the first neural network (MLP) to the Cartesian translational command $(\Delta x, \Delta y)$ required to move the robot over the target object. If the displacement is too large – therefore the parallax effects are too disturbing – the robot moves first and looks again (top loop, $r_{fine} = 4\,mm$). The rotational adjustments are determined from an (u,v)-centered region of interest (ROI with size 50×50 pixel). A small set of features $\mathbf{f}$ is extracted and a second neural network (also MLP) maps to the shortest rotational approach command $\Delta\phi$. Executing the robot move command prepares the system for the next step, which is usually the force-torque guarded grasping of the object. Alternatively, the procedure can be repeated (by closing the loop) for testing or in case of poor image conditions.

Preprocessing: The grabbed color image of size 192×144 pixels is reduced to one channel by a pixel-wise maximum selection in the R,G, and B channel. Then the mean and standard deviation of all pixel values is computed and a global linear pixel intensity transformation is applied which normalizes the image to the training standard conditions

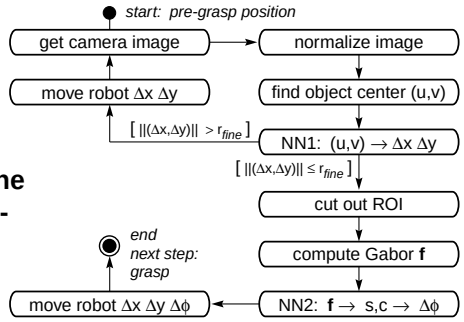


Figure 3. The perception-action loop

(i.e. the same intensity average and variance).

Object Localization: The normalized image is binarized and a standard blob detection algorithm selects the object center. If the image is not cluttered, a fast row and column-nwise histogramming is sufficient. The first neural network NN1, a 2-3-2 resilient backprop accelerated MLP, maps to the desired translational correction $(\Delta x, \Delta y)$, which is used in the Cartesian transfer command and send to the robot.

Object Rotation and Angle Wrapping: The second neural network (NN2) has to code for the shortest rotational correction $\Delta\phi$. Here occurs the problem of angle wrapping ($\phi = \phi \pm 360°$) and the rotation object symmetry count κ, e.g., for the cube $\kappa = 4$ (i.e. same appearance for $\Delta\phi = 0°$, 90°, 180°, 270°). We solve this with an angular sine/cosine pair encoding for the MLP-output layer

$$s = \sin(\kappa\Delta\phi), \quad c = \cos(\kappa\Delta\phi), \tag{3}$$

and the back-transformation

$$\Delta\phi = \frac{1}{\kappa} \operatorname{atan}\left(\frac{s}{c}\right). \tag{4}$$

Selecting the Training Data Set: The desired nominal grasping position is *"demonstrated"* to the system by guiding the robot *once* (e.g. via a 3D-mouse) in the correct pose and defining the approach distance. Starting from there, a set of images is automatically aquired. An image gets grabbed from the hand camera after displacing the robot by the value $-(\Delta x, \Delta y, \Delta\phi) \in [-a, a]\times[-a, a]\times[-b, b]$ (for NN1 $a = 25\,mm$ and NN2 $a = 5\,mm$, $b = 180°/\kappa$; the training set is sampled from a 3×3×7-grid, while the test set is randomly sampled). The image processing results in association with the desired robot command $(\Delta x, \Delta y)$ for NN1, and $\Delta\phi$ for NN2, providing the training data for the supervised learning phase of the neural networks.

Selecting the Gabor Filter Set: Since the optimal feature set for our task was unknown, we carried out some systematic simulation tests with varying Gabor filter combinations. The RMS positioning accuracy of the entire system was evaluated on 10 training and test cycles with randomized starting conditions for NN2.

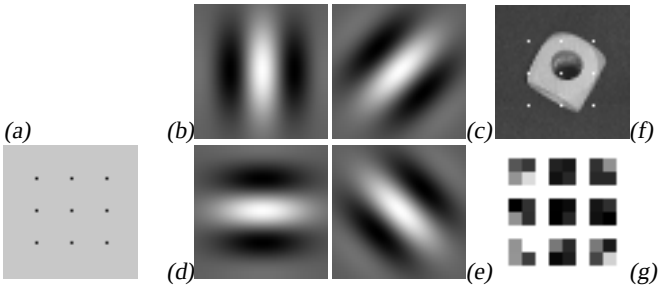


Figure 4. *(a)* Locations of the receptive fields centers (p, q) in the ROI-image. *(b-e)* Gabor filter set with 4 orientations, shown only for the middle location. The other 32 filters are shifted versions. *(f)* The ROI image with (a) superimposed. *(g)* The resulting 36 features visualized as gray image.

The winning filter set configuration was a surprise: it consists of only nine different center positions on a 3×3 grid centered in the $(50 \times 50\,pix)$ image ($p, q \in \{13, 25, 27\}\,pix$). At each position we centered four even Gabor filters with $\lambda = 30\,pix$, $\sigma = 12.5\,pix$, and $\theta \in \{0°, 45°, 90°, 135°\}$ as depicted in Fig. 4. The first surprise was, that already 36 image features are sufficient for the rotation estimation task. The second was, that the system does not prefer higher Gabor frequencies which would be more sensitive to the edge positions.

4. Experimental Results

Accuracy: We achieved an asymptotic RMS positioning accuracy of $0.08\,mm$, $0.2\,mm$ (in x, y direction) and $0.8°$ (in ϕ) after a couple of iterations in the fine-positioning loop in Fig. 3. This is far more than required. For reasonably good illumination conditions one short – pure translational (termed "half" loop) – and one full positioning loops is sufficient (half+full="$1\frac{1}{2}$"), see Tab. 1.

Robustness – Illumination: Changing local lighting conditions are a real threat to many vision based control algorithms. The described preprocessing method proved quite efficient: with stepwise dimmed lights and furthermore changing to a single sideward lamp (producing bad cast-shadows), we found that the performance degraded only in the speed of convergence. The basic operation was stable up to extremely poor illumination conditions.

Different Object Colors and Background: The training object (yellow cubical) shows very good contrast to the gray table surface. But the fine-positioning system works without modifications also for comparable objects, for other colors, and under bad conditions, e.g. poor brightness contrast and dim light. As Fig. 5 displays, partly covered objects or even textured background did not bring the grasping system to tumble.

Table 1	**Gabor System**	**PCA System** in [7]	**Hough System** in [6]
Accuracy with *Good* Illumination	(0.5, 0.7, 6.6°) after $1\frac{1}{2}$ loops		(4 DOF: x, y, z, ϕ)
$(x/mm, y/mm, \phi/°)$	(0.1, 0.3, 1.2°) after $2\frac{1}{2}$ loops	(0.4, 0.7, 0.6°)	(0.5, 0.8, 1, 1.4°)
Save Grasp requires	$1\frac{1}{2}$ loops	5 loops	$2\frac{1}{2}$ loops
Accuracy with *Poor* Illumination	(0.2, 0.2, 1°) after $6\frac{1}{2}$ loops	(3.1, 1.0, 6.1°)	—
Save Grasp requires	$3\frac{1}{2} - 5\frac{1}{2}$ loops	20 loops	*[miracle]*
Controller	2 × MLP	4 × Neuro-Fuzzy	PSOM + Model
ϕ Feature Generation by	36 general filter masks	3 eigenimages	line HT
Total Time per Loop Iteration	1–2 sec	1–2 sec	1–2 sec
Training Time	$<$ 1 min	3 hours	$<$ 1 min

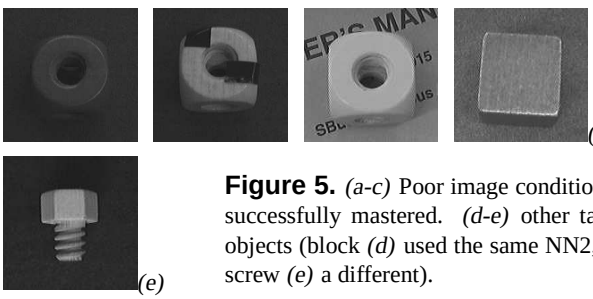
5. Discussion and Conclusion

We presented a pre-grasp fine positioning scheme for a robot camera-in-hand system. It employs a small set of only 36 Gabor masks probing the image for spacial frequency content at nine locations and four orientations. The feature set is universal for a family of similar objects and can be easily adapted and/or enriched for a broader spectrum of shapes. The preprocessing stage performs an image intensity adaptation and guidance of the rotational sensor (ROI). The overall system is robust with respect to the image condition, e.g. changes in illumination and some amount of occlusions and clutter.

Comparison With PCA-System: The PCA and neuro-fuzzy controlled approach [7] was implemented in the same lab, pursuing the same goal – but applying different techniques. Based on the appearance of the object, the *eigenimages* are computed and the image is encoded in an only three-dimensional *eigenspace* (for the rotational part). The small number is mainly a compromise to balance between the exponentially growing training time and the information gained. Of course, as more useful information the system can extract, as fewer iterations it needs for precise grasping. As Tab. 1 lists, the main disadvantage of the PCA-FC system is the limited training and performance speed. On the other hand, the PCA-system displays reliability and robustness.

Comparision With Hough-Transform (Line-HT): [6] employs classical image processing techniques. The image is taken from a vertical viewing angle, since the contour (more precise, the binarized edge-processing output) must be polygonal. The method delivers good results if (and only if) the conditions are highly normalized (good illumination, good background contrast, no clutter, etc.). The main reason for the poor robustness is the focus on differential image information (edges), the associated noise sensitivity, and the information loss in the binarization step. The countermeasures are expensive (e.g. contour following, region growing, etc.) and do not fundamentally solve the problem.

Summarizing, the Gabor-filter based system uses very favorably image processing techniques, working also in visual cortex neurons. Employing only 36 "simple cells" we built a technical system which was *(i) calibration free* (e.g. no camera calibration required), *(ii) direct* (no expensive image processing like segmentation, region growing, etc.), and *(iii) fast* (Gabor feature detection and ANN mappings could be implemented in real-time). It demonstrated *(iv)* the *robustness* real-world capable system require.



(a-d) *(e)*

Figure 5. *(a-c)* Poor image conditions – successfully mastered. *(d-e)* other target objects (block *(d)* used the same NN2, the screw *(e)* a different).

References

[1] J. Daugman. Two-dimensional spectral analysis of cortical receptive fi eld profi les. *Vision Research*, 20:847–856, 1980.

[2] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics, Image Processing*, 44:87–116, 1988.

[3] J. Jones and L. Palmer. An evaluation of the two-dimensional gabor fi lter model of simple receptive fi elds in cat striate cortex. *Journal of Neurophysiology*, 58:1233–1258, 1987.

[4] M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Computers*, 42:300–311, 1993.

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

[6] D. Schwammkrug, J. Walter, and H. Ritter. Rapid learning of robot grasping positions. In *Proc. 7th Int. Symp. Intelligent Robotic Sys (SIRS)*, pages 149–155, July 1999.

[7] J. Zhang, R. Schmidt, and A. Knoll. Appearance-based visual learning in a neuro-fuzzy model for fi ne-positioning of manipulators. In *Proc. IEEE ICRA*, 1999.