APPEARANCE–BASED NEURAL IMAGE PROCESSING FOR 3–D OBJECT RECOGNITION AND LOCALIZATION

C. Yuan

Fraunhofer Institute FIT
Schloss Birlinghoven
53754 Sankt Augustin, Germany

H. Niemann

Chair for Pattern Recognition
University of Erlangen–Nürnberg
Martensstr. 3, 91058 Erlangen, Germany

ABSTRACT

This paper presents an appearance–based neural image processing system for the recognition and localization of 3–D objects. All the objects are placed on the table and can be moved arbitrarily around, allowing both in–plane and out–of–plane rotations. Instead of doing object segmentation and object specific geometric modeling, objects are directly modeled by their appearance. First a principal component network is configured to generate a nonlinear filter. Then the objects are represented in a feature vector derived by hierarchical nonlinear filtering of the input image. With this compact feature vector and a small set of training samples, a neural classifier is configured for recognition purpose. Based on the same feature vector, object is localized by several neural pose estimators. Results for the recognition and localization of a large number of real images under heterogeneous background are shown.

1. INTRODUCTION

Recognition of 3–D objects from arbitrary viewpoint is a fundamental issue in computer vision and has applications in many areas. To identify an object in a changing environment is challenging since besides the background problem objects themselves vary appearance in an image under different viewing and illumination conditions. Hence it is particular important to model object appearance.

As one of the most successful approach in image classification and object recognition [8], appearance–based object recognition has gained wide attention. It applies an embedded information representation and takes into account object variations under different viewing conditions. Most of the appearance–based approaches are based on subspace methods, e.g. the eigenface approach proposed by [4], multidimensional receptive field histograms used by [7], Gabor and wavelet transform in [5] and other transforms such as [3]. Most of the available appearance–based approaches utilize a bounding box around the object in an image. As object appearance in an image varies significantly under external rotations, it is difficult to keep the size of the bounding box fixed. The result is either two much background information in the object bounding box or necessary interpolations must be done, which costs extra CPU time for doing image processing and degrades recognition performance [6].

In order to overcome the above problem, we developed an integrated neural appearance–based approach for the representation, recognition and localization of 3–D objects in 2–D images. Our algorithm employs neither object segmentation nor bounding box. Object representation is done implicitly based on the whole information content of an image. And the object model is built through an automatic learning phase. As brightness variations caused by the 3–D shape, surface reflectance properties, texture, illumination conditions and so on are all encoded in the integrated neural appearance model, recognition can be done using the maximal information.

2. OBJECT REPRESENTATION

In image processing and pattern recognition, a widely used method for extracting low–dimensional manifolds is based on the Principal Component Analysis (PCA). Its nonlinear variation, the nonlinear PCA is particularly useful, as it can provide more accurate representation by building a nonlinear (curved) lower–dimensional surface called principal surface that "passes through the middle of the data" [2]. Suppose functions $\boldsymbol{g}$ and $\boldsymbol{h}$ are two nonlinear mapping each from $\Re^n$ to $\Re^m$ and from $\Re^m$ to $\Re^n$ respectively, the target of the nonlinear PCA is the minimization of the mean squared error in nonlinear reconstruction

$$J = E||\boldsymbol{x} - \boldsymbol{h}(\boldsymbol{g}(\boldsymbol{x}))||^2 \tag{1}$$

by an optimal choice of $\boldsymbol{g}$ and $\boldsymbol{h}$.

In this paper, we developed a nonlinear Principal Component Networks (PCN) for the realization of the nonlinear principal manifolds. As shown in Fig. 1 (a), such network is a kind of autoassociative network whose output is trained

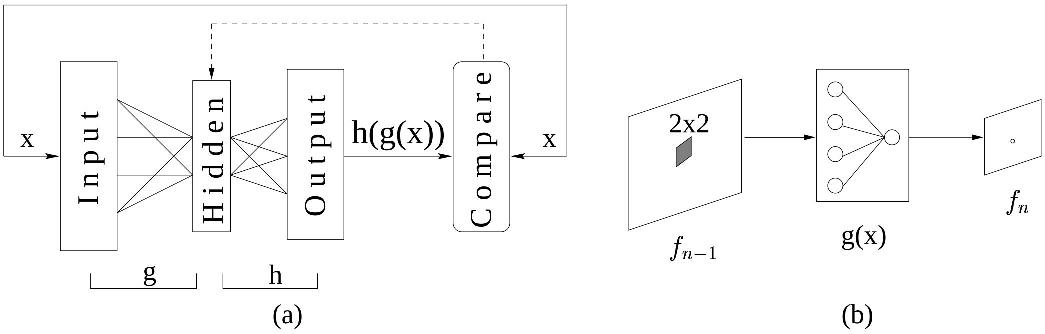

Fig. 1. Principal Component Network

to reproduce the input $\boldsymbol{x}$. And the hidden unit activations correspond to a feature vector in $\Re^m$. The mapping from the input layer to hidden layer and from the hidden layer to the output layer can be regarded as the nonlinear function $\boldsymbol{g}$ and $\boldsymbol{h}$, respectively.

A 4–1–4 PCN has been trained to obtain the nonlinear function $\boldsymbol{g}$. During the training process the network receives 2 × 2 subimages which are cut sequentially out of the input image $\boldsymbol{f}_0$ (256×256 pixels) without overlap. The result function $\boldsymbol{g}$ is similar to that of a lowpass filter:

$$\boldsymbol{g}(\boldsymbol{x}) = \mathcal{T}\left(\sum_{k=1}^{4}(w_k x_k + \theta)\right) \qquad (2)$$

where w_k, θ are the weight vector and threshold of the hidden neuron and $\mathcal{T}(s)$ is the sigmoid activation function:

$$\mathcal{T}(s) = \frac{1}{1 + exp(-s)}. \qquad (3)$$

Apply the function $\boldsymbol{g}$ one time, we get an image $\boldsymbol{f}_1$ which is one–fourth of the original image $\boldsymbol{f}_0$ (see Fig. 1 (b)). By repeating such operation

$$\boldsymbol{f}_n(i,j) = \boldsymbol{g} * \boldsymbol{f}_{n-1}(i,j) \qquad (4)$$

four times, we get a feature vector $\boldsymbol{c} = \boldsymbol{f}_4$ of 256 (16 × 16) dimensions, which can be used subsequently in the recognition and localization process.

3. OBJECT RECOGNITION

The identification of multiple 3–D objects can be viewed as a mapping from a set of input variables represented by $\boldsymbol{c} = \boldsymbol{f}_4$ to a set of output variables representing the class labels. Suppose the output variables are denoted by y_j, with $j = 1, \ldots, \lambda$. The mapping can be modeled in terms of some mathematical functions which contain a number of adjustable parameters: $y_j = y_j(\boldsymbol{c}; \boldsymbol{w})$, where $\boldsymbol{w}$ is a vector which embraces in it the parameters whose value can be determined with the help of the training data. A three layer network whose number of input neurons is equal to the dimension of **c** and whose number of output neurons is equal to λ can be applied to form a model for the classification. As verified in [1], the activation of the j–th output neuron, y_j, can be interpreted as measuring the *a posteriori* probability function $P(\Omega_j|\boldsymbol{c})$ for class Ω_j. According to Bayes rule, the object represented by vector $\boldsymbol{c}$ should be classified as coming from class Ω_κ with

$$\kappa = \underset{j}{\mathrm{argmax}}\{y_j\}. \qquad (5)$$

4. OBJECT LOCALIZATION

Object localization is the process of determining the position of the identified objects with respect to a coordinate system, or in short object pose estimation. For a 3–D object, its pose parameter space is six–dimensional, which consists of the rotation

$$\boldsymbol{R} = \boldsymbol{R}_z \boldsymbol{R}_y \boldsymbol{R}_x \in \Re^{3\times 3} \qquad (6)$$

and the translation

$$\boldsymbol{t} = (t_x, t_y, t_z)^{\mathrm{T}} \in \Re^3. \qquad (7)$$

Here $\boldsymbol{R}_x$, $\boldsymbol{R}_y$, $\boldsymbol{R}_z$ are rotation matrices with rotation angle ϕ_x, ϕ_y and ϕ_z around the x–, y– and z–axis, respectively. Since we use a fix mounted camera, t_z is a constant. Hence there is no need to estimate this parameter.

4.1. Translation parameter estimation

For estimating the translation parameters t_x and t_y, a network consisted of two neural estimators is configured. As shown in Fig. 2, each translation estimator has one hidden layer (hidden neurons are illustrated as squares), with each hidden neuron having only horizontal or vertical connection to the input. The activation of the two output neurons T_x and

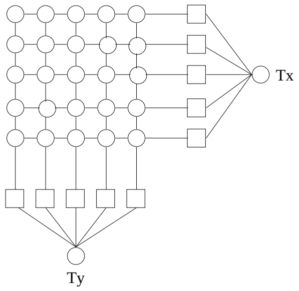

Fig. 2. Translation parameter estimation.

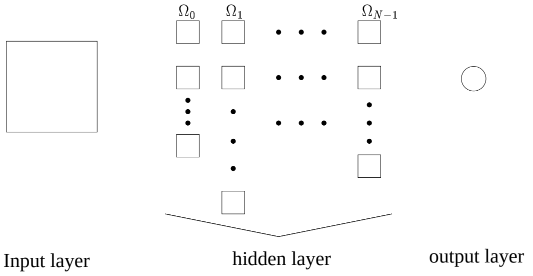

Fig. 3. Illustration of the trained DLVQ net.

T_y is between 0 and 1, and each of them is computed as:

$$T_x = \mathcal{T}\left(\sum_j w_j \mathcal{T}\left(\sum_i w_{ij}\boldsymbol{c}(x_i,y_j)\right)\right) \quad (8)$$

$$T_y = \mathcal{T}\left(\sum_i w_i \mathcal{T}\left(\sum_j w_{ji}\boldsymbol{c}(x_i,y_j)\right)\right) \quad (9)$$

where $\boldsymbol{c}(x_i,y_j)$ is the feature vector that the two estimators receive, w_{ij} and w_{ji} are the hidden neuron weights, and w_j and w_i the output neuron weight parameter of the two neural estimators, respectively. $\mathcal{T}$ is the the same sigmoid activation function as given in equation (3).

Now object position can be estimated by multiplying T_x and T_y with the width (D_x) and height (D_y) of the input image $\boldsymbol{f}_0$, respectively:

$$t_x = D_x \times T_x \quad (10)$$

$$t_y = D_y \times T_y \quad (11)$$

4.2. Rotation parameter estimation

Unlike classification and translation parameter estimation, rotation parameters cannot be simply estimated by a network which can output a real value representing the rotation angles directly. Yet by a redefinition of the problem, it can be solved as follows: Since vectors belonging to the same class should have smaller difference than those belonging to different classes, discrepancy between similar views of an object should be also smaller than the difference between diverse views. This leads to the solution of estimating rotation angles by trying to find a natural grouping of the images. Consequently, the problem of rotation parameter estimation becomes the classification of different views into different classes.

Here for each of the rotation parameters ϕ_x, ϕ_y and ϕ_z, a dynamic learning vector quantization (DLVQ) net [9] as shown in Fig. 3 is built. Each DLVQ net receives the same feature vector and has only one output neuron. At initialization stage, the number of neurons in the hidden layer N is equal to the number of possible rotation angles. By a sampling interval $\Delta = 3^o$ for possible rotation within 360^o, there should be $N = 120$ hidden neurons at initialization stage. With the DLVQ net, rotation parameters can be estimated by classifying the feature vector of the input images into $N = 120$ classes $\Omega_0, \Omega_1, \ldots, \Omega_{N-1}$.

Training of the net is based on the DLVQ algorithm, with which a natural grouping in the data can be found. Suppose that the vector $\boldsymbol{c}_\lambda$ belonging to the same class (with the same perspective view) are distributed normally with a mean vector $\boldsymbol{\mu}_\lambda$. A feature vector $\boldsymbol{c}$ should then be assigned to the class Ω_κ, to which it has the smallest Euclidean distance ($||\boldsymbol{\mu}_\kappa - \boldsymbol{c}|| = \text{argmin}_\lambda\{||\boldsymbol{\mu}_\lambda - \boldsymbol{c}||\}$). During the learning phase, the algorithm computes a new mean vector $\boldsymbol{\mu}_\lambda$ for each class once every cycle and generates the hidden layer dynamically. Training is finished when the correct $\boldsymbol{\mu}_\lambda$ is found for every class. Because each view of an object has different statistics, the generated hidden neuron numbers in each class can vary, which is illustrated in Fig. 3. After training, the network can output a natural number n ($0 \leq n < N$) based on new input. The rotation angle is computed as to be equal to $n \times \Delta$.

5. EXPERIMENTAL RESULTS

For the evaluation of the approach, a set of fourteen objects is used, which can be seen from Fig. 4. The images are taken with a CCD–camera mounted on a robot arm. Through the moving and turning of the objects and the tilting of the robot arm it is possible to generate images of these objects with different pose parameters. And during the image capture process, 3 illumination conditions are arranged. Also on some images the background is not homogeneous. Two sequences of data are available. The first sequence has 3720 images for each of the object. For half of the 14 objects, a second sequence is available. And there are 3600 images for each of the seven objects. But the viewing angles and object locations of the second sequence are totally different from the first one. Altogether there are 52080 images in the first sequence and 25200 images in the second sequence.

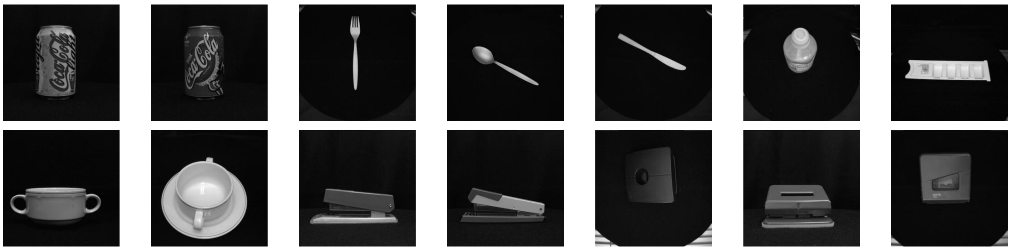

Fig. 4. Objects used in experiments.

Test set	Recognition rate(%)	Translation error (pixel)	Rotation error (^{o})
Sequence 1	95.3	1.4	1.9
Sequence 2	95.1	1.2	1.6
Average	95.2	1.3	1.8

Table 1. Experimental results

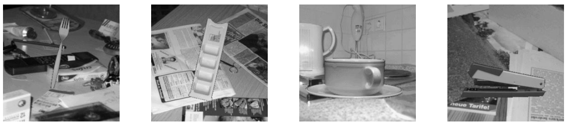

Fig. 5. Images with complex background

We chose only 300 images/class from the first sequence for training. And all the rest of sequence 1 and the whole of sequence 2 are used for test purpose.

It takes about 45 ms for the system to recognize and localize an input image on a Linux PC with 2.0 GHz processor. Listed in Table 1 is the achieved recognition and localization results on the test sets, which are disjoint from the training set. The average recognition rate achieved is 95.2%, with an average localization accuracy of 1.3 pixel for translation and 1.8^o for rotation. The robustness of the neural appearance based approach is demonstrated from the fact that there is no big difference between the two results on sequence 1 and sequence 2.

The proposed approach can also be applied to images with more complex background, as is shown in Fig. 5. For each of the fourteen objects, we collected 3720 such images and used one half of them for training and the other half for test. An average of 72.5% recognition rate is obtained on the test set.

6. CONCLUSION

A new image processing framework for recognition and localization of 3–D objects using a single 2–D image was presented, where neural networks are widely applied to model the object appearance. It doesn't require any explicit modeling of object geometry and hence can be applied arbitrarily to any 3–D objects. Extensive experimental study involving a large set of images of fourteen objects with complex background shows the effectiveness and accuracy of the proposed approach. Despite the fact that some objects look very similar and are difficult to differentiate, the system performs quite well and achieves promising recognition rate and localization accuracy. In the future, we will apply this approach for multiple object tracking from video image sequences.

7. REFERENCES

[1] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford Clarendon Press, 1995.

[2] K.I. Diamantaras and S.Y. Kung. *Principal Component Neural Networks: Theory and Applications*. Wiley, New York, 1996.

[3] J. Hornegger, H. Niemann, and R. Risack. Appearance-based object recognition using optimal feature transforms. *Pattern Recognition*, 33:209–224, 2000.

[4] H. Murase and S.K. Nayar. Visual learning and recognition of 3–D objects from appearance. *IJCV*, 14(1):5–24, 1995.

[5] J. Poesl and H. Niemann. Wavelet features for statistical object localization without segmentation. In *ICIP97*, pages 170–173, 1997.

[6] M. Reinhold, D. Paulus, and H. Niemann. Improved appearance-based 3-D object recognition using wavelet features. In T. Ertl, B. Girod, G. Greiner, H. Niemann, and H.-P. Seidel, editors, *Vision, Modeling, and Visualization 2001*, pages 473–480, Stuttgart, November 2001. AKA/IOS Press, Berlin, Amsterdam.

[7] B. Schiele and J.L. Crowley. Probabilistic object recognition using multidimensional receptive field histograms. In *ICPR96*, pages 50–54, 1996.

[8] A. Selinger and R.C. Nelson. Improving appearance–based object recognition in cluttered background. In *ICPR*, volume 1, pages 46–50, September 2000.

[9] A. Zell. *Simulation neuronaler Netze*. Addison-Wesley, Bonn, 1994.