IAPR Workshop on CV - Special Hardware and Industrial Applications OCT.12-14, 1988, Tokyo

A NEW SCHEME FOR PRACTICAL FLEXIBLE AND INTELLIGENT VISION SYSTEMS

Nobuyuki OTSU and Takio KURITA

Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba-shi, Ibaraki-ken, Japan 305

ABSTRACT

We propose a new scheme for practical vision systems which are simple in structure, directly and adaptively trainable for various purposes. The feature extraction consists of two stages: At first, general and primitive features which are shift-invariant and additive are extracted over the retinal image plane. Then those features are linearly combined on the bases of multivariate analysis methods so as to provide new effective features for each task. Such a system can adaptively and automatically learn the task from the given supervised training examples and show an intelligent performance. Some examples of practical applications to object recognition and measurements are experimentally shown.

1 INTRODUCTION

Recently there are increasing needs for practical application of computer vision in various fields, including industrial or biomedical inspection, measurement, assortment (recognition), and so on. Especially, such vision systems that are convenient (personal computer based), practical (on-line, real-time) and, if possible, multi-purpose (flexible, adaptively trainable), have been receiving considerable attention.

Such requirements seem to start revealing a discrepancy of the conventional approach to computer vision via the steps of image processing techniques.

To make the point clear, let us consider a simple example. Suppose a binary image in which two kinds of circles with different diameters scatter without overlapping, and a task is to count the numbers of the two kinds of circles in the image. A conventional approach we tend to adopt will be as follows: First, all the objects (circles) are detected and labeled. Then each object is checked one at a time (presumably by measuring the diameter somehow) and classified. Each decision is counted up, and we get the final counts. Such a sequential procedure requires steps of complicated techniques of image processing, and obviously the computation time linearly grows as the number of the circles increases, and real-time processing becomes prohibitive. It is also noted that such a system lacks adaptability since the techniques are "given" only for the given task.

This indicates the need of more direct and new methods; in other words, the need of change of paradigm from "sequential and procedural vision" to "parallel and adaptive (trainable) vision".

Reflecting on the above viewpoint, we propose in this paper a new scheme for practical vision systems, which is simple in structure and directly and adaptively trainable for various purposes. The basic idea is similar to Perceptron [1] or neural networks (for example, see [2]) but more practical being based on closed-form solutions for statistical performance criteria without employing such slow iterative learning processes.

The basic idea of the system has already been presented in [3] and registered as patents (for example, [4]). A hardware implementation (named AISS) has been developed in cooperation with OKK Inc. and the detail will be presented in a different paper [5] together with actual demonstration at this conference.

2 GENERAL FORMULATION

Many of the tasks required for practical vision systems can be characterized by the following examples of questions and answers made on the object(s) in a retinal image plane.

Q1: What is this? (recognition of a single object)

A1: That is a "A".

Q2: How much is the area? (measurement)

A2: It is "n" pixels (or mm^2).

Q3: What are these? (recognition of multiple objects)

A3: Those are one "A" and two "B's".

Q4: How many "A's" are there? (identification and counting)

A4: There are "m" ("A's").

As seen in the example in the previous section, the conventional approach of "sequential and procedural vision" requires too special and tedious steps of image processing algorithms, which makes it difficult to design adaptive and real-time systems. We would like to design a simple and flexible system which can answer in real time to such questions as in the above (even for various "A" and "m"). In order to do this, we had better reconsider a general frame of feature extraction from the standpoint of "parallel and adaptive vision".

Feature extraction in pattern recognition is generally consists of two stages; Geometrical Feature Extraction and Statistical Feature Extraction. The

first stage (GFE) concerns the extraction of features which are *invariant* under some transformation group acting on patterns, and the second stage (SFE) concerns how to combine those invariant features into *effective* features for recognition. An intensive theoretical study on these aspects has been done in [6].

Viewing the above questions and answers in this frame, the most important points are: 1) the statements are basically irrelevant to where the objects locate in the retinal plane, 2) the answer m (also n) is metric and should be proportional to the real number of objects (or to the area), and of course 3) a system should be flexible and adaptive to such a variety of questions. Therefore, the essential requirements for the feature extraction are:

1) To be shift-invariant

2) To be additive

3) To be adaptively trainable.

Being based on these considerations, we propose the following general frame of feature extraction which consists of two stages.

S1: General and primitive features which are shift-invariant and additive are extracted over the retinal image plane, satisfying the requirement 1) and 2).

S2: Those primitive features are linearly combined on the bases of multivariate analysis methods so as to provide new effective features for each task, satisfying the requirement 3).

Such a system can adaptively and automatically learn the task from the given supervised training examples (images).

In Section 3, we present the primitive features adopted in S1. Actual examples of the adaptive system (S1+S2) are described in Section 4, for general recognition purpose by using Discriminant Analysis and for general measurement purpose by using Multiple Regression Analysis.

3 PRIMITIVE FEATURES

Let a retinal plane be denoted by P, which is, for example, a rectangular region in two-dimensional plane R^2; $P \subset R^2$. Images on P are represented by functions $f(\boldsymbol{r}) \geq 0$ defined within P, where $\boldsymbol{r} \in P$ and the support of f; $Supp(f) = \{\boldsymbol{r} | f(\boldsymbol{r}) > 0\}$, is included in P; $Supp(f) \subset P$.

A shift (translation) of $f(\boldsymbol{r})$ within P is represented by

$$T(\boldsymbol{a})f(\boldsymbol{r}) = f(\boldsymbol{r} + \boldsymbol{a}),$$

where the displacement $\boldsymbol{a} \in R^2$ is restricted such that the support does not exceed P.

Let $x[f]$ denote a feature of image $f(\boldsymbol{r})$ extracted over P. Then, the requirement 1); viz., $x[f]$ is shift-invariant, is represented by

$$x[T(\boldsymbol{a})f] = x[f] \quad \text{for} \quad \forall \boldsymbol{a};\ Supp\,(T(\boldsymbol{a})f) \subset P.$$

On the other hand, the requirement 2); viz. $x[f]$ is additive, is represented by

$$x[f_1 + f_2] = x[f_1] + x[f_2] \quad \text{for} \quad Supp(f_1) \cap Supp(f_2) = \emptyset.$$

The requirements 1) and 2) lead us to such features that are given by sums of local features over P.

It is well known that the autocorrelation function is shift-invariant. Its extension to higher orders has been presented in [8]: For N displacements $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$, the Nth-order autocorrelation function is defined by

$$r_f^N(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N) = \int_P f(\boldsymbol{r}) f(\boldsymbol{r} + \boldsymbol{a}_1) \cdots f(\boldsymbol{r} + \boldsymbol{a}_N) d\boldsymbol{r}.$$

Our primary concern is the shapes of objects on the retinal plane P. Therefore, we shall restrict images $f(\boldsymbol{r})$ to binary images which cover a wide range of practical application. Thus, gray-level images taken by a TV (CCD) camera are supposed to be thresholded into binary images (as to a method of automatic thresholding, see [7]).

Then, for binary images the Nth-order autocorrelation function can be regarded as counting the number of pixels which satisfy some logical condition; namely,

$$f(\boldsymbol{r}) \wedge f(\boldsymbol{r} + \boldsymbol{a}_1) \wedge \cdots \wedge f(\boldsymbol{r} + \boldsymbol{a}_N) = 1,$$

therefore in other words it is counting the patterns characterized by the above logical statement over the binary image f. Obviously, the scan by the reference point $\boldsymbol{r}$ can be restricted to the "black" pixels of image; $f(\boldsymbol{r}) = 1$.

Since the number of these autocorrelation functions obtained by the combination of the displacements are enormous, we must reduce it for practical application. First, we restrict the order N up to two ($N = 0, 1, 2$). It is seen that the 0th-order autocorrelation is just counting the number of black pixels of a binary image f. We also restrict the range of the displacements to within a local 3×3 region, the center of which is the reference point (**Fig. 1-a**). The number of the patterns of so restricted the displacements is $2^8 = 256$, since the center pixel can be fixed to 1 (black) as mentioned above. However, it is redundant, because it doubly counts the patterns which are equivalent in the shift in the scanning. By further elimination of the equivalent displacements we have 25 essentially different patterns, which we called the masks (See [3] for details). The masks are shown in **Fig. 1-b**, where the symbol * represents "don't care".

Primitive features x_j are obtained by scanning the binary image over P with the 25 local masks M_j. The features are obviously shift-invariant and also additive for isolated objects on P, satisfying our primary requirements 1) and 2). For on-line and real-time vision systems, we can use a special hardware (AISS) which extracts these primitive features in real time.

4 ADAPTIVE SYSTEMS

An adaptive vision system can be constructed on the basis of the new scheme of the general frame of

feature extraction, S1 plus S2. Problem is how to use (handle, combine) those 25 primitive features extracted in S1. Here, the requirement 3), adaptability and trainability, should be considered. This leads us to the use of multivariate data analysis methods for the second stage S2.

In multivariate data analysis methods, new features y_i are given by linear combinations of the primitive features, with weights a_{ij} being coefficients:

$$y_i = \sum_{j=1}^{M} a_{ij} x_j \quad (i = 1, \ldots, N),$$

or simply in vector-matrix notation,

$$y = A'x,$$

where $A = [a_{ij}]$ is the coefficient matrix, the symbol $'$ denotes the transpose, and M is now equal to 25. Then, the optimal parameters, the weights in A, are determined so as to optimize a criterion function $\Gamma(A; \Pi)$ which is set to evaluate the performance of the linear model for a given task, where Π is the statistics of x defined from the learning samples.

This frame for S1 looks like the Perceptron or the neural nets in PDP but differs in that the latter ones assume the nonlinearity of y_i for outputs, for example thresholding or sigmoid function. Such a condition as restricting outputs and also inputs to 0 or 1 or in-between apparently comes from the analogy to the neurons. But, it is irrelevant to our scheme. Our aim is not to analyze nor simulate the performance of neural nets. The linearity is important for the requirement 2), otherwise the additive property is destroyed. Also, the linearity makes the learning so simple, resulting in a closed form solution. This is a big difference from the nonlinear cases where we have to resort to an iterative learning process like the error back propagation algorithm.

It is usually said that a linear model is not enough, however it is not necessarily so as will be seen in the actual experimental results shown later. Whether a linear model or not, it depends on application domain and is a trade-off in computational complexity.

4.1 Recognition System

A general purpose recognition system is constructed by using Discriminant Analysis (DA) for the second stage feature extraction S2 and by combining it to a classifier (for details, see [3]). New features y_i which effectively discriminate the classes of patterns are extracted from the primitive features x_j as follows.

Suppose we have K pattern classes $\{C_i\}_{i=1}^{K}$. The within-class and the between-class covariance matrices of x are computed from training sample images as

$$X_W = \sum_{i=1}^{K} \omega_i X_i, \quad X_B = \sum_{i=1}^{K} \omega_i (\overline{x}_i - \overline{x}_T)(\overline{x}_i - \overline{x}_T)',$$

where ω_i, $\overline{x}_i$, $\overline{x}_T$, and X_i denote *a priori* probability of class C_i, the mean vector of class C_i, the total mean vector, and the covariance matrix of class C_i, respectively. The discriminant criterion is to maximize $\mathrm{tr}(Y_W^{-1} Y_B)$, where Y_W and Y_B are the within- and the between-class covariance matrices defined similarly on y. The optimal coefficient matrix A is then given by the solution of the following eigen equation:

$$X_B A = X_W A \Lambda, \quad A' X_W A = I,$$

where Λ is a diagonal matrix of eigenvalues and I denotes the unit matrix. The j-th column of A is the eigen vector corresponding to the j-th largest eigenvalue. Thus, the N new features y_j are evaluated in its importance for discrimination. The maximum number N is bounded by $\min(K-1, 25)$.

In **Fig. 2**, we show the result of practical application to character recognition. Each of the training data and the test data consists of 2,220 samples of alpha-numeric characters type-printed with a used ribbon; 60 samples each for 37 characters (A-Z, 0-9, Yen symbol). The retinal plane P is 36 × 36 in meshes. **Fig. 2-a** shows some examples of the samples. The recognition rate is illustrated in **Fig. 2-b**, where n in the abscissa means that the first n new features are used. The real line shows the result for the training data, and the break line shows the result for test data. The classifier we used is the simple one which checks the distances from an input (y) to class means ($\overline{y}_i$) and classifies the input to such class C_j that gives the least distance. The result is quite good, attaining to 98.7% for training data and 98.5 % for test data at $n = 10$. It should be noticed that we never teach the system what features should be taken. It is also noted that the character position within P does not matter.

Another experiment was conducted as a recognition system which classifies acute angled triangles and obtuse angled triangles. We showed the system randomly generated 28 triangles three times with different rotations (84 learning samples in total), from which new discriminant features were automatically extracted. The training samples were fed to the system again as test samples. The system made mistakes only for almost right angled triangles. The averaged error rate was 3.57% (**Table 1**). The rate will be reduced by increasing the number of training samples.

More interesting experiments will be conducted for recognition purposes where we hardly imagine what features should be extracted to discriminate pattern classes (for example see [5]).

4.2 Measurement System

A general purpose measurement system is also constructed by using Multiple Regression Analysis (MRA) for the second stage feature extraction S2. In this case we need no classifier. The linear model $y = A'x$ directly estimates the measurement vector z that is really observed for a given image. The dimension N of z and y indicates the number of measurements observed simultaneously. Various kinds of

measurements can be considered for z_j; for examples, area or perimeter of an object within retinal plane P, or even the number of isolated objects within P. Thus, the application as a measurement system is quite general and wide. An application of MRA to adaptive and automatic design of image processing filters has been shown in [9].

The coefficient matrix A in the model is determined so as to minimize the mean square error between the desired measurement vector $\boldsymbol{z}$ and the estimated vector $\boldsymbol{y}$ for a set F of training sample images:

$$\varepsilon^2(A) = \mathop{\mathrm{E}}_{F} \| \boldsymbol{y} - \boldsymbol{z} \|^2 = \mathop{\mathrm{E}}_{F} \| A'\boldsymbol{x} - \boldsymbol{z} \|^2,$$

where E stands for expectation (or arithmetic mean). The closed form solution is given by

$$A = R_{xx}^{-1} R_{xz}$$

where R_{xx}=E($\boldsymbol{x}\boldsymbol{x}'$) is the autocorrelation matrix of $\boldsymbol{x}$, and R_{xz}=E($\boldsymbol{x}\boldsymbol{z}'$) is the crosscorelation matrix between $\boldsymbol{x}$ and $\boldsymbol{z}$.

An application was done to the case of $N = 2$; measurement of the numbers of two sizes of particles within P, which was exemplified in Introduction. We artificially generated 40 binary images each of which contains some numbers of circles with two different diameters. The training samples were shown to the system together with the correct answers. In **Fig. 3-a** through **3-c** we show the examples of the generated images. **Fig. 3-d** is the actual binary image of **3-c** taken into the system via the TV camera and thresholding. In spite of such noisy images, the system learns the task and after that can estimate the numbers quite correctly.

More interesting examples of application are the measurements of topological characteristics (invariants) of binary images; for example, the number of isolated objects or the number of holes in an object within P. It should be noticed that the numbers are irrelevant to the shapes of the objects or of the holes. Experiments were done for the number of objects (without holes) by using 48 generated sample images (examples are shown in **Fig. 4-a** and **4-b**) and for the number of holes by using 42 generated sample images (examples are shown in **Fig. 4-c** and **4-d**). The system learnt how to estimate such a number from the training sample images and could answer quite correctly.

Such a possibility has already been suggested in [1] and studied in [10], where the local patterns (2×2 masks) to compute such a number have been designed on the basis of the Euler formula. As a matter of fact, the meshes of a digital binary image can be regarded as a triangulation of the image (a two-dimensional complex). The direction of the approach in our system is upside down; namely, the system learns (finds) by itself the Euler formula from the training samples. This is the interesting and important point of the present system.

Generally speaking, if a real answer is included in the model, then the system can learn the correct answer. Theoretically, it is possible only by showing different training samples the number of which is equal to that of the parameters (25) in the model. An obvious case is the estimation of the area (pixels) of an image. In fact the number is given by x_1.

5 CONCLUSION

We proposed a new scheme for practical and adaptively trainable vision systems from the standpoint of parallel computation. The system can adaptively learn by itself the task from the given training samples. Unlike the slow iterative learning of Perceptron and neural nets, the learning is very fast due to the closed form solution by the multivariate analysis.

In spite of its simplicity and generality, the system has a high potentiality as an adaptive vision system as well as its wide range of practical applications.

ACKNOWLEDGMENTS

The authors are indebted to the colleagues of the Mathematical Engineering Section for helpful discussions and to Dr. Akio Tojo, the director of the Computer Science Division, for his encouragement. They express their gratitude to Mr's S. Kuwashima and T. Shibata of OKK Inc. for helping them to use AISS.

References

[1] M. Minsky and S. Papert, *Perceptrons*, MIT Press (1969).

[2] D.E. Rumelhart, J.L. McClelland, and The PDP Research Group, *Parallel Distributed Processing*, MIT Press (1986).

[3] N. Otsu, T. Shimada and S. Mori, "A method of pictorial feature extraction based on Nth-order autocorrelation masks," (in Japanese) IECE Tech. Rep., PRL 78-31 (1978).

[4] N. Otsu, S. Mori and T. Saito, "Method and apparatus for character reading," U.S. patent 4288779 (Sep. 1981).

[5] T. Shibata and S. Kuwashima, "An approach for real-time pattern recognition of images by AISS," Proc. IAPR Workshop on Computer Vision (to appear) (1988).

[6] N. Otsu, *Mathematical Studies on Feature Extraction in Pattern Recognition*, (in Japanese) Researches of the Electrotechnical Laboratory, No. 818 (1981).

[7] N. Otsu, "Discriminant and least squares threshold selection," Proc. 4th Int. Joint Conf. on Pattern Recognition, 592–596 (Nov. 1978).

[8] J. A. Mclaughlin and J. Raviv, "Nth-order autocorrelations in pattern recognition," Information and Control, **12**, 121–142 (1968).

[9] N. Otsu, "A multiple regression analysis approach to the automatic design of adaptive image processing systems," Proc. SPIE, **435**, 70–75 (Aug. 1983).

[10] S.B. Gray, "Local properties of binary images in two dimensions," IEEE Trans. on Computers, **C-20**, 551–561 (1971).

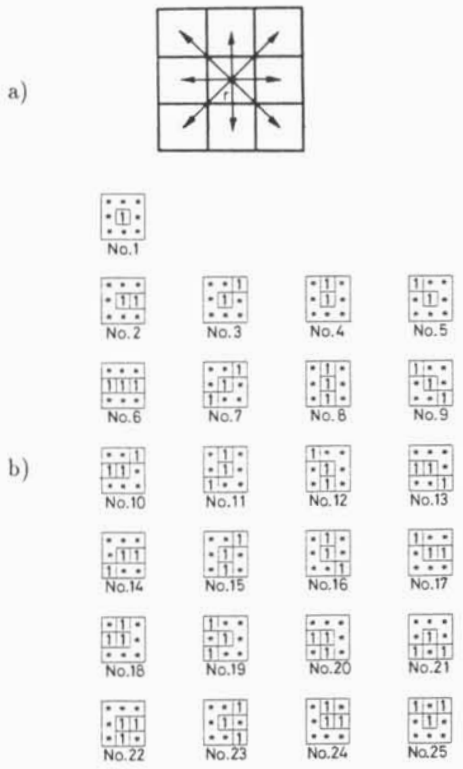

Figure 1: Primitive feature extraction. a) The directions of the displacements. b) Mask patterns.

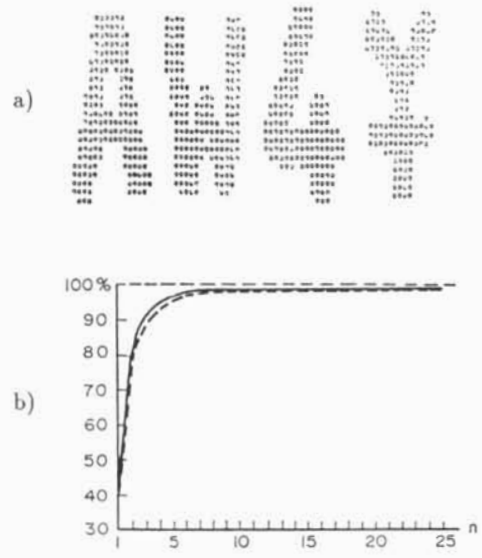

Figure 2: Recognition of A&N characters. a) Examples of samples. b) Recognition rate.

Table 1. Recognition of triangles.

	Correct	False	Error Rate (%)
Acute (33)	32	1	3.03
Obtuse (51)	49	2	3.92
Total (84)	81	3	3.57

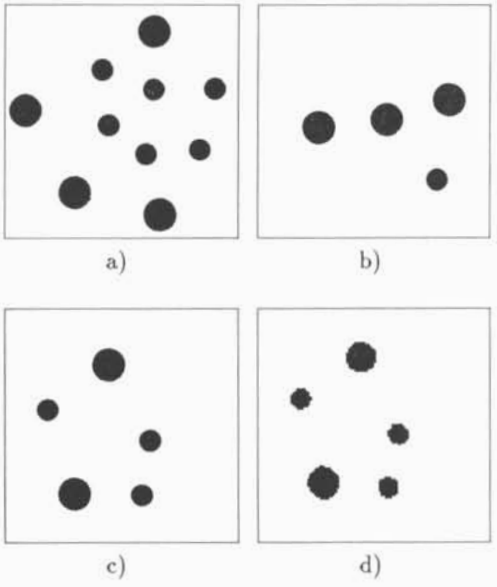

Figure 3: Measurement of the numbers of two sizes of circles. Estimated numbers N for large circles and n for small circles are $N = 4.10$, $n = 5.88$ for a), $N = 3.09$, $n = 0.88$ for b), and $N = 1.97$, $n = 3.02$ for c) and d).

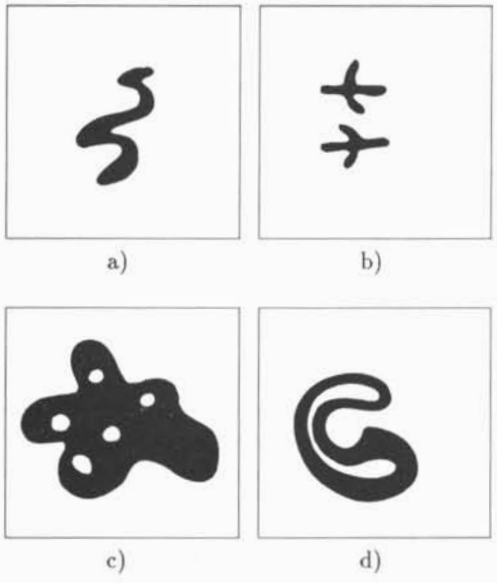

Figure 4: Measurements of topological characteristics. a) and b) are examples of images used for the measurement of the number of objects, and c) and d) are for the measurement of the number of holes.