Open issues in pattern recognition

Robert P.W. Duin[1] and Elżbieta Pękalska[1,2]

[1] ICT group, Faculty of Electr. Eng., Mathematics and Computer Science
Delft University of Technology, The Netherlands
{r.p.w.duin,e.pekalska}@ewi.tudelft.nl
[2] School of Computer Science, University of Manchester, United Kingdom
pekalska@cs.man.ac.uk

Summary. The area of pattern recognition has developed itself into a mature engineering field with many practical applications. This increased applicability, together with the development of sensors and computer resources, leads to new research areas and raises new questions. In this paper, old and new open issues are discussed that have to be faced in advancing real world applications. Some may only be overcome by brute force procedures, while others may be solved or circumvented either by novel and better procedures, or by a better understanding of their causes. Here, we will try to identify a number of open issues and define them as well as possible.

1 Introduction

Pattern recognition is the human ability to see regularities in observations. From the early development of computers, scientists and engineers tried to imitate this ability by mechanical means, either partially or in its entirety. Two main types of results have been obtained from these efforts so far.

First, a better understanding is reached of the human perception, reasoning and the ability to gain new knowledge and to apply it to a changing environment. This knowledge is partially formulated in physical and biological terms, giving more insight into the study of human senses and the neural system. To some extent, this knowledge is also partially expressed in mental, psychological and epistemological terms, describing how facts and observations are combined by reasoning, how uncertainty is handled and how conclusions are reached. Attempts to design sensors, computers and programs that simulate or mimic these processes bring an additional prospect to the investigation of possible biological models. An ever returning difficulty, however, is the relation of low level phenomena occurring in the senses and the nerves to a high level understanding and conceptual thinking.

Second, various pattern recognition systems have been developed that are of practical use, as for the assistance in medical diagnosis, industrial inspection, personal identification and man-machine interaction. Very often, they

are not based on a detailed simulation of the human processes, but on independent approaches to the problem at hand.

In this paper, we will focus on the pattern recognition research aiming at the development of automatic systems as discussed above. We will especially deal with the possibilities of these systems to learn from sets of examples. We will also consider the process of going from a low level of single objects observed by sensors to a higher level of decision making, based on the global pattern of a class of objects. As already mentioned above, it is still little understood how this emergent process develops on the human level. Although some technical solutions and explanations exist, there is still the intuitive feeling that these are far from optimal. The basic question here asks how incidental observations, suffering from noise and considered in a particular context, can be integrated into general knowledge about classes of objects, independent of the noisy observations and the accidental circumstances.

A number of open issues will be discussed in the area of automatic pattern recognition. This is the field in which the development of recognition systems that learn from examples is studied. For some recent references, see the books of Webb [29] and Van der Heijden et al. [16] and the review by Jain et al. [18]. The more expert knowledge on the field of application is integrated, the better such a system is going to be. However, the ability to learn often conflicts with the implementation of detailed physical knowledge, as the first relies on flexibility, while the latter tries to reduce that. For this reason, we will focus on systems for statistical pattern recognition as they are concerned with learning from observations.

The issues to be described are just a selection of the many points which are not yet entirely understood. Some of them may be solved in the future by the development of novel procedures or by gaining an additional understanding. Others may remain an issue of concern to be dealt with in each application separately. In the subsequent sections, we will systematically describe them according to the following line of the advancement of a pattern recognition system:

- *Representation.* This is the way individual real world objects and phenomena are numerically described (or encoded) such that they can be related to each other in some meaningful mathematical framework. This framework has to allow the generalization to take place.
- *Design set.* This is the set of objects available or selected to develop the recognition system.
- *Adaptation.* This is usually a reduction of the representation such that it becomes more suitable for the generalization step.
- *Generalization.* This is the step in which objects of the design set are related such that classes of objects can be distinguished and new objects can be accurately classified.
- *Evaluation.* This is an estimate of the performance of a developed recognition system.

2 Representation

The problem of representation is a core issue for pattern recognition [5, 7]. It encodes the real world objects by some numerical description, handled by computers in such a way that the individual object representations can be inter-related. Based on that, later a generalization is achieved, establishing descriptions or discriminations between classes of objects. Originally, the issue of representation was almost neglected, as it was reduced to the demand of having good features provided by some expert. The learning is often believed to start at the given feature vector space. Indeed, many books on pattern recognition disregard the topic of representation, simply by assuming that objects are somehow already represented [2, 25].

A systematic study on representation [9, 24] is not easy, as it is application- or domain-dependent (where the word 'domain' refers to the nature or character of problems and the resulting type of data). For instance, the representations of a time signal, an image of an isolated 2D object, an image of a set of objects placed on some background, a 3D object reconstruction or the collected set of outcomes of a medical examination are entirely different observations that need separate approaches to find good representations. Anyway, if the starting point of a pattern recognition problem is not well defined, this cannot be improved later in the process of learning. It is, therefore, of crucial importance to study the representation issues seriously. Some of them are phrased in the subsequent sections.

2.1 The use of vector spaces

Traditionally, objects are represented by vectors in a feature vector space. This representation makes it very feasible to perform some generalization (with respect to this linear space), e.g. by estimating density functions for classes of objects. However, the object structure is lost in such a description. If objects contain an inherent, identifiable structure or organization, then relations between their elements, like relations between neighboring pixels in an image, are entirely neglected. This also holds for spatial properties such as Fourier coefficients or wavelets weights. These original structures might be partially re-discovered by deriving statistics over a set of vectors (representing objects), however, these are not included in the representation itself. One may wonder whether the representation of objects as vectors in a space in not too oversimplified to be able to reflect the nature of objects in a proper way. Perhaps objects might be better represented by convex bodies in a space or by some other structures. The generalization over sets of vectors, however, is heavily studied and mathematically well developed. How to generalize over a set of other structures is still an open question.

The essential problem of the use of vector spaces for object representation is originally pointed out by Goldfarb [13]. He prefers a structural representation in which the original object organization (connectedness of building

structural elements) is preserved. However, as a generalization procedure for structural representations does not exist yet, Goldfarb starts from the evolving transformation systems [12] to develop a novel system [14].

Issue: How to overcome the fundamental inadequacy of vector space representations?

2.2 Compactness

An important, but seldom explicitly identified property of representations is compactness [1]. In order to consider classes, which are bounded in their domains, the representation should be constraint: objects that are similar in reality should be close in their representations. If this demand is not satisfied, objects may be described arbitrarily. Hence, there is no generalization.

This so-called *compactness hypothesis* puts some restriction on the possible probability density functions that classes may have in a vector space used for the representation, e.g. the feature space. This, thereby, also narrows the set of possible classification problems. A formal description of the probability distribution of this set may be of interest to estimate the expected performance of classification procedures for an arbitrary problem.

The *no-free-lunch theorem* claims that all expected performances for all classifiers are equal (in particular equal to the random assignment rule) [30]. This pessimistic result is based on an unbounded set of possible problems, not limited by the compactness hypothesis. If the latter is taken into account, the study on generalization abilities may be significantly improved.

Issue: What is the distribution of classification problems that is in agreement with the compactness hypothesis?

2.3 Representation types

The following representations are here distinguished:

- *Features.* Objects are described by a set of characteristic attributes. If these attributes are continuous, the representation is usually compact. Nominal and categorical attributes may cause problems. As a description by features is a reduction of objects to vectors, different objects may have the same representation. Consequently, classes may overlap.
- *Pixels* or other samples. A complete representation of an object may be approximated by its sampling. For images, these are pixels, for time signals, these are time samples and for spectra, these are wavelengths. A pixel representation is a specific, boundary case of a feature representation, as it describes the object properties in each point of observation.
- *Probability models.* Object characteristics may be related by some probabilistic model. Such models may be based on expert knowledge or trained from examples. Mixtures of knowledge and probability estimates are difficult, especially for larger models.

- *Structural models.* Instead of using probabilities, object models may also be based on a structural description. Automatic procedures aiming at the design of such descriptions from a set of examples are still in their childhood and mainly restricted to the estimation of model parameters. How to learn a structure is not yet clear. Some ideas can be found in [14].
- *Dissimilarities.* Instead of an absolute description by features, objects are relatively described by their dissimilarities to a collection of specified objects. These may be carefully selected prototypes, but also random subsets of the training set may work well [23]. The dissimilarities may be derived from raw data, such as images, spectra or time samples, from original feature representations or from structural representations such as strings or relational graphs. If the dissimilarity measure is nonnegative and zero only for two identical objects, always belonging to the same class, the class overlap may be avoided by dissimilarity representations.
- *Similarities.* In contrast to dissimilarities, similarities may be naturally additive with respect to the support (characteristics) for particular classes. Attributes that are in agreement with some class membership may increase the similarity to that class. For this reason, a similarity representation may be good to deal with missing values and partially characterized objects.
- *Conceptual representation.* Objects may be related to classes in various ways, e.g. by a set of classifiers, each based on a different representation, training set or model. The combined set of these initial classifications or clusterings constitute a new representation [24]. This is used in the area of combining clusterings [10] or combining classifiers [20].

Object descriptions in feature spaces and by dissimilarity representations constitute a good basis for generalization in some appropriately determined spaces. It is, however, difficult to integrate them with the detailed prior knowledge that one has on classes. On the other hand, probabilistic models and structural models, especially, are well suitable for this integration. They, however, constitute a weak basis for training general classification schemes. Usually, they are limited to assign objects to the class model that fits best based on the nearest neighbor rule.

Issue: Can representations be found that offer a good basis for modeling object structure and which can also be used for generalizing from examples?

2.4 Missing data problem

Recognition of partially characterized objects is important for many applications. Depending on the representation, many solutions are investigated, often trying to estimate some values for the missing data item. Here, we will just emphasize again the possibility of using a similarity representation for approaching this problem. Instead of estimating the missing values, it may be worth using the data example that are available.

Issue: Can similarities solve the missing data problem?

2.5 Optimized and trainable representations

The representation may be based on background knowledge of the recognition problem at hand. Some representations, like the conceptual one, are based on a generalization over other representations. Below we will discuss the adaptation of a representation to the optimal conditions following from a generalization procedure. In addition, it may be also possible to learn from the raw data what a good representation is, independently of the problem knowledge and independently of the generalization procedure.

Concerning the learning process, two types of representations are considered: optimized (or fixed) ones and trainable ones; see also [7, 24]. Optimized representations rely on some initial representations for which specific parameters are to be found. For instance, for dissimilarity representations, the measure itself is assumed to be given. It can be optimized with respect to a set of objects, but rather in a limited way such as the specification of a nonlinear transformation and the search for optimal parameters. This is also related to the adaptation step discussed below.

Trainable representations should be built on raw measurements and rely on some identified collection of sub-patterns that may be used to learn to describe the internal structure of objects. In the process of an active design, particular sub-patterns should be chosen together with some weights such that each class separately possesses a compact description and is well defined in the presence of other classes. This may be judged with respect to the chosen classification procedure. Another possibility to build a trainable representation is to consider a conceptual representation, based e.g. on a proximity of an object to a class. This proximity is related to the costs (weights of transformations) of generating an object from a set of primitives (basic descriptors) in the context of other objects within a class, as well as objects outside this class. Such an attempt is carried out in [14], where not only the essential transformations and the weights are learnt, but structural primitives (sub-patterns) as well.

Issue: Can good representations be learnt?

2.6 Spatial connectivity

In general, recognition problems may be context dependent. If this context is not incorporated to the representation (which often occurs in practice), the resulting conflicts have to be solved afterwards. An example is an image recognition system using pixels as features (hence a 16×16 image is represented as a point in a 256-dimensional space). The spatial connectivity between the neighboring pixels is not preserved in such a feature representation. It may be retrieved from the correlations between the features (pixels), but it is not included in the representation itself. Consequently, a statistical decision function built on this representation neglects the original structure of an image. This is inherent to the vector space inadequacy observed by

Goldfarb [13]. Feasible approaches incorporating the spatial connectivity to numerical representations have to be still developed. A possibility might be offered by proximity (similarity or dissimilarity) representations derived from intermediate structural description of objects [7, 24].

Issue: How to incorporate contextual relations into the representation?

3 Design Set

A pattern recognition problem is not only defined by a representation itself, but also by the set of examples given for training and evaluating a classifier in its various stages. The selection of this set and its usage strongly influence the overall performance of the final system. We will discuss some related issues.

3.1 Multiple use of the training set

The total design set or its parts are used in several stages during the development of a recognition system. Usually, one starts from the exploration of this set, which may lead to the removal of wrongly scanned or erroneously labeled objects. After gaining some insights into the problem, the analyst may select a classification procedure based on his/her observations. Next, the set of objects may go through some normalization. Additionally, the representation has to be optimized, e.g. by a feature selection or extraction algorithm. Then, a series of classifiers has to be trained and the best ones need to be selected or combined. An overall evaluation may result in a re-iteration of some steps leading to different choices.

In the entire process the same objects may be used a number of times for the estimation, training, validation, selection and evaluation. Usually, one estimates an average error by a cross-validation. It is well known that the multiple use of objects should be avoided as it biases the results and decisions. Re-using objects, however, is almost unavoidable in practice. A general theory about how much a training set is 'worn-out' by its use and which compensations or corrections may be possible does not exist, yet.

Issue: What is a general theory on the re-use of datasets for training?

3.2 Representativeness of the training set

Training sets should be representative for the objects to be classified by the final system. Usually, a randomly selected subset of the latter is used for training. Intuitively, it seems to be useless to collect many objects represented in the regions where classes do not overlap. On the contrary, in the proximity of the decision boundary, depending on its complexity (non-linearity) and the class overlap, a higher sampling rate seems to be advantageous. This is, of course, inherently related to the chosen classification procedure.

Such a representativeness can be only discussed for static problems, i.e. where raw measurements used to define representations do not significantly change over time. To rephrase it, we assume that the circumstances of the measurement collecting process are stable or if they change, the variable factors are identifiable and their influence on the construction of the final representation is negligible. In other situations, one should consider an active approach, where a classifier develops in time.

Issue: When is the training set sufficiently well sampled and representative for the recognition problem? Should a classifier develop over time?

3.3 Unknown or undetermined class distributions

For some problems, like in medical or machine diagnostics cases, the object distributions for one or more classes are badly defined or even undetermined. For instance, how the class of non-faces can be defined in the face detection problem? Or in machine diagnostics, what is the probability distribution of all casual events if the machine will be used for undetermined production purposes? Therefore, a training set that is representative for the class distributions cannot be found. An alternative may be to sample the domain of the classes such that all possible objects are approximately covered. This means that for any object that could be encountered in practice there exists a sufficiently similar object in the training set. 'Sufficiently similar' has to be defined in relation to the specified class differences. Moreover, as class density estimates can not be derived for such a training set, class posterior probabilities cannot be computed. For this reason such a type of domain based sampling is only appropriate for non-overlapping classes. In particular, this problem is of interest for non-overlapping (dis)similarity based representations [7].

Issue: Is domain sampling possible? Can from a given dissimilarity matrix be determined whether the sampling is sufficiently dense?

4 Adaptation

Once a recognition problem has been formulated by a set of example objects in some representation, the generalization over his set may be considered, finally leading to a recognition system. However, the selection of a proper generalization procedure may not be evident, or several mismatches may exist between the realized representation and the preferred generalization procedures. This occurs when e.g. the chosen representation needs a non-linear classifier and only linear decision functions are computationally feasible, or the space dimensionality is much too high with respect to the cardinality of the training set, or the representation cannot be perfectly embedded in a Euclidean space, while most classifiers demand that. For reasons like these, various adaptations

of the representation may be considered. When class differences are explicitly preserved or emphasized, such an adaptation may be considered as a part of the generalization procedure. Some adaptation issues that are less connected to classification are discussed below.

4.1 Problem complexity

In order to determine which classification procedures might be beneficial for a given problem, Ho and Basu [17] proposed to investigate its *complexity*. This is yet an ill-defined concept. Some of its aspects include data organization, sampling, irreducibility (or redundancy) and the interplay between local and global character of the representation and/or of the classifier. Perhaps several other attributes are needed to define complexity such that it can be used to indicate a suitable pattern recognition solution to a given problem.

Issue: How can the complexity of a recognition problem be characterized?

4.2 Selection or combining

Representations may be complex, e.g. if objects are represented by a large amount of features or if they are related to a large set of prototype examples. A collection of classifiers can be designed to make use of this fact and later combined; see section 5. Additionally, also a number of representations may be considered simultaneously. In all these situations, the question arises whether a selection has to be made from the various sources of information, or whether some type of combination should be preferred. A selection may be made randomly, or be based on a systematic search procedure for which many strategies and criteria are possible. Combinations may sometimes be fixed, e.g. by taking an average, or a type of a parameterized combination like a weighted linear combination as a principal component analysis (PCA); see also [3, 21, 24].

The choice for some selection or combining procedure is sometimes dictated by economical arguments, minimizing the amount of necessary measurements or computations. If this is not an issue, the decision has to be made based on accuracy arguments. Selection neglects some information, while combination tries to use everything. The latter, however, may suffer from overtraining as weights or other parameters have to be estimated and may be adapted to the noise in the data. The recently popular sparse solutions offered by support vector machines [26] and sparse linear programming approaches [15, 11] constitute a way of compromise. How to optimize them is not yet clear.

Issue: What are the advantageous ways of optimizing a set of representations?

4.3 Nonlinear transformations

One way to build a simple classifier such as a linear function for a representation that demands a more complicated, nonlinear solution is to transform the representation in an appropriate way to emphasize the linear aspects. One special example is the transformation of a non-Euclidean dissimilarity representation such that it becomes embeddable in a Euclidean space. However, such a nonlinear transformation is not directly focused on finding the best classifier, as it just prepares the framework for future generalization. See [22] for a discussion. It is, thereby, doubtful whether this two-step procedure is better than a direct use of a nonlinear classifier on the original representation.

Issue: When are nonlinear transformations of the representation useful?

4.4 Class structure or class distribution

Assume we have some high-dimensional vector representation of a recognition problem. The training set consists of a set of vectors for each class. Do these vectors constitute a cloud of points, like we tend to draw on a piece of paper if we explain classification procedures in 2D spaces? This idea probably does not hold in high-dimensional spaces. Many representations simply do not posses as many degrees of freedom as the space dimensionality. Their intrinsic dimensionality tends to be much lower. Also a linear subspace, as e.g. the one found by the PCA, is often not a good model of a class description. The reason is that the linear interpolation between two vectors, representing objects of the same class, may produce representations for which no objects exist in reality. For instance, a linear interpolation between two images of different faces does not usually create a proper face image. Consequently, the class of face images, even of the same person with a slightly rotated head positions, yields neither a cloud of points, nor a linear subspace in some representation space. Most likely, classes constitute nonlinear manifolds in these high-dimensional spaces.

Issue: How to constitute classifiers making use of the fact that classes constitute non-linear structures when represented in high-dimensional spaces?

5 Generalization

The generalization over sets of points leading to class descriptions or discriminants was extensively studied in pattern recognition in the 60's and 70's of the previous century. Many classifiers were designed, based on the assumption of normal distributions, kernels or potential functions, nearest neighbor rules, multi-layer perceptrons, etcetera [4, 29, 18]. These types of studies were later extended by the fields of multivariate statistics, artificial neural networks and machine learning. However, in the pattern recognition community, there is still a high interest in the classification problem, especially in relation to practical

questions concerning issues of combining classifiers, novelty detection or the handling of ill-sampled classes.

5.1 Classifier selection or classifier combination

The issue of selection or combining holds for representations as well as for classification. In the latter case, it is more clear and apparent as no further steps have to be considered. If a set of classifiers is computed in the process of developing a recognition system, do we take the best one, or do we combine them? Which analysis should produce the right answer? For selection some performance estimation is needed, e.g. based on an evaluation set, or on a systematic cross-validation scheme. Why is it not possible to use the same scheme for combining? See [20] for some ideas.

Issue: How to decide between a selection or combining a set of classifiers?
Issue: Which are good sets of classifiers to be combined?

5.2 Trained or fixed combining

A set of well trained classifiers may yield the classification outputs that are further compared and combined by a fixed procedure like averaging or product [19]. Imperfectly trained classifiers may be combined by a trained combiner. For this an additional training set is needed that may also be used to train the base classifiers better. If the same training set is used for training both the combiner and the base classifiers, then training of the combiner will suffer from bias and will not be representative for new objects, unless the base classifiers are undertrained. In this case, however, a fixed combiner may work well too. Consequently, it is not clear when and how a combiner should be trained [6].

Issue: When and how should a combining classifier be trained?

5.3 Sequential or parallel training

The architecture of a combined classifier with linear base classifiers is very similar to a neural network. The main difference refers to the training step, either classifier by classifier, or as an entire system at once. A similar difference exists between the feature selection followed by a linear classifier and training a sparse linear classifier that reduces the feature space by its design. Many more examples exist, like a PCA followed by a classifier or a regularized classifier. Is it possible to find a general rule that gives insights when either sequential or parallel training has to be preferred?

Issue: When should a recognition system be trained part by part and when in its entirety?

5.4 Classifier typology

Any classification procedure has its own explicit or built-in assumptions with respect to the class distributions or other characteristics. This implies that for a problem that exactly fulfils the assumption of a particular procedure, this procedure will generate a relatively well-performing classifier. Consequently, any classification approach has its problem for which it is the best. In some cases such a problem might be far from reality. The construction of such problems may reveal which the typical characteristics of a particular procedure are. Moreover, when new proposals are to be evaluated, it may be demanded that some examples of its corresponding typical classification problem are published, making clear what the area of application may be. See also [8].

Issue: A library of problems corresponding to the library of classifiers.

5.5 Generalization principles

What is the basic principle of generalization over a set of examples? How can we apply some rules to identify an unobserved property of an object that is different from all objects in the given set of examples? The two basic generalization principles are probabilistic inference, using the Bayes' rule and the minimum description length principle that determines the most simple model in agreement with the observations (Occam's razor). These two principles are essentially different. The first one is sensitive to multiple copies of an existing object in the training set, while the second one is not. Consequently, the latter is not based on densities, but just on object differences or distances. When should each principle be followed? Should this be decided from the start, in the selection of the design set and the way of building a representation, or is it possible to postpone it for later?

Issue: Bayes or Occam?

5.6 The use of unlabeled objects and active learning

The above mentioned principles are examples of statistical inductive learning, where a classifier is induced based on the design set and it is later applied to unknown objects. The disadvantage of such approach is that a decision function is in fact designed for all possible representations, whether valid or not. Transductive learning is an appealing alternative as it determines the class membership only for the objects in question, while relying on the collected design set or its suitable subset.

The use of unlabeled objects, not just the one to be classified, is a general principle that may be applied in many situations. It may improve a classifier based on just a labeled training set. If this is understood properly, the classification of an entire test set may yield better results than the classification of object by object.

Issue: How to make use of unlabeled data to construct classifiers?

5.7 Multi-class problems

Two-class problems constitute the traditional basic line in pattern recognition. It boils down to finding a discriminant or a binary decision. Multi-class problems can be formulated either as a series of two-class problems (this can be done in various ways, none of them is entirely satisfactory) or as a detection problem in which each of the possible classes is searched for. The one that fits best is used. This approach neglects all alternatives in the first step, but compares them later.

Issue: Should multi-class recognition be performed by detection or by classification?

5.8 One-class problems

In contrast, but also very similar to multi-class problems, are the one-class problems. In the latter case, a single class is well defined and all alternatives, outliers, classes (of any number) are just ill-sampled, not sampled at all or undefined. Discriminants seem to be inappropriate, while the class detectors do not use what might be available on the alternatives. Densities cannot be applied properly as they cannot be estimated for the outlier class; see [27, 28]. One-class classifiers can be constructed as examples of proximity-based conceptual representations [7, 24].

Issue: What is a proper one-class classifier?

5.9 Domain based classification

There are several reasons why densities cannot be used (properly) for constructing classifiers. Any classifier based on averages like the computation of a mean square error assumes a distribution of the training objects in agreement with a class distribution. If this does not exist, or if the training set does not agree with it, such classifiers cannot be formally used. As explained above, an appealing alternative is the use of distances and/or the minimum description length principle. Classifiers based on this cannot decide in class overlapping regions. So they need a representation that avoids this. The dissimilarity representation may be suitable, however, others may be used as well if the ground is found justifying why an overlap is avoided. So, there is a need for domain based classifiers [7]. These are classifiers that assume no class overlap and that do not require that the distribution of the training set follows the class distribution. The class domain, however, should still be sampled properly in such a way that it can be approximated or used to construct a decision function.

Issue: How to construct domain-based classifiers.

5.10 Object structure

We repeat here the issue already raised in the section on representation. Suppose that the object structure is incorporated to the representation. How do we generalize over a set of examples? Does this result describe a structure, a distribution, some domain in a space or a growth model? Again we refer to the proposal of Goldfarb [14] which seems to be a very general attempt to solve this problem.

Issue: How to learn the structure in objects?

6 Evaluation

Two questions are always apparent in the development of recognition systems. These are: how good is a particular system once it is trained and which are good recognition procedures in general? The first question has sometimes a definite answer, while the second one is open.

6.1 Recognition system performance

What do we mean if we wonder how good a system is? Is it its accuracy on average, computed over all objects we are going to classify or is it determined by the worst error it may be made? In the first case, we again assume that the set of objects to be recognized is well defined (in terms of distributions). Then, it can be sampled and the accuracy of the entire system can be estimated based on an evaluation set. We now neglect the issue that after using this evaluation set together with the training set, a better system can be found. A more interesting point is how to judge the performance of a system if the distribution of objects is ill-defined or if a domain based classification system is used as discussed above. Now, the largest mistake made becomes a crucial factor for this type of judgements. One needs to be careful, however, as this may refer to an unimportant outlier (resulting e.g. from invalid measurements).

Issue: How to evaluate the performance for ill-defined class distributions?

6.2 Prior probability of problems

How good is a recognition procedure in general? As argued above, any procedure has a problem for which it performs well. So, how large is the class of such problems? We cannot state that any classifier is better than any other classifier, unless the distribution of problems to which these classifiers will be applied is defined. Such distributions are hardly studied. What is done at most is that classifiers are compared over a collection of benchmark problems. Such sets are usually defined ad hoc and just serve as an illustration. The set of problems to which a classification procedure will be applied is not defined.

Issue: How to judge the expected performance of a recognition procedure?

7 Discussion and conclusions

Pattern recognition is a human activity that we try to imitate by mechanical means. There are no physical laws that assign observations to classes. It is the human consciousness that groups observations together. Although their connections and inter-relations are often hidden, by the attempt of imitating this process, some understanding might be gained. The human process of learning patterns from examples may follow along the lines of trial and error. It has, however, to be strongly doubted whether statistics play an important role in this process. Estimating probabilities, especially in multi-variate situations is not very intuitive for majority of people. Moreover, the large amount of examples needed to build a reliable classifier by statistical means is much larger than it is available for human learning.

In human recognition, proximities based on relations between objects seem to come before features are searched and may be, thereby, more fundamental. For this reason and the above observation we think that the study of dissimilarities, distances and domain based classifiers are of great interest. This is further encouraged by the fact that such representations offer a bridge between the possibilities of learning in vector spaces and the structural description of objects that preserve relations between object's inherent structure.

We think that the use of dissimilarities for representation, generalization and evaluation constitute the most intriguing issues in pattern recognition.

References

1. Arkedev AG and Braverman EM (1966) Computers and Pattern Recognition. Thompson. Washington, DC.
2. Bishop CM (1995), Neural Networks for Pattern Recognition. Clarendon Press.
3. de Diego IM, Moguerza JM, and Muñoz A (2004) Combining Kernel Information for Support Vector Classification. Multiple Classifier Systems. LNCS:3077. Springer-Verlag. 102-111.
4. Duda RO, Hart PE and Stork DG (2001). Pattern Classification 2nd. edition, John Wiley & Sons.
5. Duin RPW, Roli F, and De Ridder D (2002). A note on core research issues for statistical pattern recognition, Pattern Recognition Letters, 23:493-499.
6. Duin RPW (2002), The Combining Classifier: To Train Or Not To Train? ICPR2002 II:765-770.
7. Duin RPW, Pękalska E, Paclík P, and Tax DMJ (2004). The dissimilarity representation, a basis for domain based pattern recognition?, In: Pattern representation and the future of pattern recognition, Workshop ICPR2004. 43-56.
8. Duin RPW, Pękalska E, and Tax DMJ (2004) The characterization of classification problems by classifier disagreements. Proc. ICPR II:140-143.
9. Edelman S (1999), Representation and Recognition in Vision, MIT Press.
10. Fred A, and Jain AK (2002), Data clustering using evidence accumulation, ICPR2002. Quebec City, Canada. 276-280.

11. Fung, GM and Mangasarian OL (2004), A Feature Selection Newton Method for Support Vector Machine Classification. Computational Optimization and Aplications 28:185202.
12. Goldfarb L, (1990), On the foundations of intelligent processes – I. An evolving model for pattern recognition. Pattern Recognition. 23(6)595-616.
13. Goldfarb L, and Hook J (1998), Why classical models for pattern recognition are not pattern recognition models. In: International Conference on Advances in Pattern Recognition. Springer. 405-414.
14. Goldfarb L, Gay D, Golubitsky O, and Korkin D (2004), What is a structural representation? 2nd version. Faculty of Computer Science, UNB. Technical Report TR04-165.
15. Graepel T, Herbrich R, Schölkopf B, Smola A, Bartlett P, Müller KR, Obermayer K and Williamson R (1999) Classification on Proximity Data with LP-Machines. ICANN 1991, 304-309.
16. Van der Heijden F, Duin RPW, de Ridder D and Tax DMJ (2004) Classification, Parameter Estimation and State Estimation. An Engineering Approach using Matlab. John Wiley & Sons Ltd.
17. Ho TK, and Basu M (2002) Complexity measures of supervised classification problems. IEEE T-PAMI 24:289-300.
18. Jain AK, Duin RPW and Mao J (2000) Statistical Pattern Recognition: A Review. IEEE T-PAMI 22:4-37.
19. Kittler J, Hatef M, Duin RPW, and Matas J (1998) On Combining Classifiers, IEEE T-PAMI 20:226-239.
20. Kuncheva LI (2004) Combining Pattern Classifiers: Methods and Algorithms. Wiley. New York.
21. Pękalska E, Skurichina M, and Duin RPW (2004) Combining Dissimilarity Representations in One-class Classifier Problems. Multiple Classifier Systems. LNCS:3077. Springer-Verlag. 122-133.
22. Pękalska E, Duin RPW, Gunter S, and Bunke H (2004) On not making dissimilarities Euclidean. In: Structural, Syntactic, and Statistical Pattern Recognition. LNCS:3138. Springer Verlag, Berlin. 1145-1154.
23. Pękalska E, Duin RPW, and Paclík P (2004) Prototype Selection forDissimilarity-based Classifiers. Pattern Recognition. accepted.
24. Pękalska E (2005) Dissimilarity representations in pattern recognition. Concepts, theory and applications. PhD thesis. Delft University of Technology.
25. Ripley BD (1996) Pattern Recognition and Neural Networks. Cambridge University Press. Cambridge.
26. Shawe-Taylor J and Cristianini N (2004) Kernel Methods for Pattern Analysis. Cambridge University Press.
27. Tax DMJ (2001) One-class classification. PhD thesis. Delft Univ. of Technology.
28. Tax DMJ, and Duin RPW (2004) Support vector data description. Machine Learning 54(1):45-56.
29. Webb A (2002) Statistical pattern recognition. Wiley. New York.
30. Wolpert DH (1995) The Mathematics of Generalization. Addison-Wesley.