DOI 10.34658/9788366741102.13

Case-Based Reasoning for Pattern Recognition using Granular Information Generated by Active Hypercontours

Piotr S. Szczepaniak[0000−0002−9973−0673]

Lodz University of Technology
Institute of Information Technology
Wólczańska 215, 90-924 Łódź, Poland

Abstract. *This paper presents a novel extension of the case-based reasoning (CBR) technique. In the proposed method, a case is defined using the concept of multidimensional information granule created by the active hypercontour. The granularity of information is expected to be justifiable. The paper explains the main principles of the method and discusses its usefulness with reference to the pattern recognition problem.*
Keywords: *pattern recognition, case-based reasoning, case-based classification, active hypercontours, information granules, hypercontour-based case specification.*

1. Introduction

Case-based reasoning (CBR) is a problem-solving approach that can be applied in a wide variety of domains. It consists in retrieving previously solved problems, which are memorized and stored as cases in a case library. The retrieved solution (case) is then adapted and reused to solve a new, similar problem. After the solution has been positively revised, such a case is included in the case library of the CBR system. This experience-based problem solving technique, and its variant called case-based classification (CBC), can be successfully applied to a range of problems, including pattern recognition.

On the other hand, contextual classifiers can be realized by sophisticated active contour methods, which can operate in multidimensional spaces. The high-level information is incorporated in an objective function, called energy. It is used for the evaluation of the quality of a hypercontour generated by the method. The search is performed in an evolution process (optimisation).

The present study endeavours to combine case-based reasoning and active hypercontours to exploit the full potential of both approaches. To achieve this goal, the case needs to be defined using the results of the active hypercontour-based method. The potential active hypercontour approach has been chosen due to its flexibility. However, this property may turn out an obstacle to correct definition of the similarity measure, which together with the definition of case constitute the two key elements of CBR (CBC). In this work, the hypercontour is applied as method of information granulation, and the information granule constitutes the case. In the literature, the concept of information granularity is considered in the context of intelligent analysis of numerical data – the discussion is usually related to the task of proximity-based fuzzy clustering. The term “granular computing” encompasses all theories, methodologies and techniques that deal with the processing of information granules.

The literature relevant to the topic of this work is given in the subsequent sections. The paper is organised as follows. First, in Section 2, the principle of case-based reasoning is presented. In Section 3, parametrized multidimensional clustering and classification realized using the potential active contour approach is described. Next, the new hypercontour-based case specification is introduced, and the role of justifiable granularity is stressed. Finally, the last section gives a brief summary of the findings.

2. CBR for Pattern Recognition

Pattern recognition is the automated discovery of regularities in data or in information granules. A pattern recognition system typically relies on three operations [1]: data or information acquisition, feature selection or extraction, and classification or clustering [2]. The first operation involves data gathering with a set of sensors. Then, the dimensionality is reduced by feature selection and extraction of information granules. This second operation significantly influences the recognition process and the quality of the results. The decision about the method is crucial, and here the use of active hypercontours is proposed because they make beneficial use of both the analysed data and the knowledge of the user (cf. Section 3). Finally, the third phase involves the processes of classification or clustering, which are performed by the subsystem that makes the decision – here, the CBR system. The more general the representation of granular information, the more sense it makes to apply case-based reasoning [3, 4].

The case-based problem solving technique, commonly referred to as *case-based reasoning* (CBR), is based on using previous experiences to understand and solve new problems. The main idea of CBR is to use a collection of past problem solutions and adapt them to address latest problems (cases) – Fig. 1.
Notation:

- S – the universe of all objects;

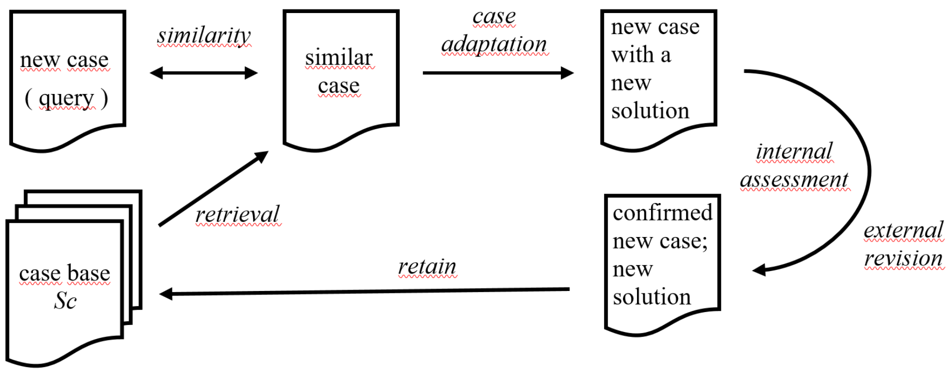

Figure 1. A model of CBR system

- *Sc* – the case base;
- c_i – the *i*-th case, $i \in I$;
- *sim* – similarity (formal definition is the appropriate approach);
- *dis* – distance;
- *p* – problem;
- *sol* – solution;
- *eff* – effect.

A base of previously solved cases (*Sc*) is given, where a case is considered as an ordered pair:

$$c_i = (p, sol)_i \tag{1}$$

The CBR process starts with the retrieval of the case which is most similar to the current one (according to the selection criterion). This reflects the idea that a solution of a similar problem may be reused to solve a new problem.

The following situations are possible: (a) the features of both entire cases – the new (query) and the candidate ones – are compared, or (b) relevant, significant portions of cases are considered.

The retrieved solution is either accepted as it is or adapted to the new problem. This process is called case adaptation (Fig. 1). The candidate solution undergoes validation, which comprises two steps: internal assessment and external validation, called revision. The latter provides definitive evidence for the correctness or practical usefulness of the candidate solution – *confirmed solution*. A case is added to the case library if it is recognized as a new and solved one.

The application domain, the users' needs, the available data, knowledge about the domain and methods, all these and many more factors determine how the CBR system will be developed, implemented and used.

The key concept in the CBR approach is *case*. As stated in Section 2, *case* is an ordered pair of problem description p and the related solution *sol*.

There are four main types of CBR that differ from one another in the manner of case representation [5]: numerical (vectors of numbers), textual, conversational, and structural.

In the pure *numerical* representation, the CBR system is often reduced to the Case-Based Classifier (CBC) [6], and information granules take the form of attribute-value vectors which include relevant information about objects, problems and solutions.

In the *textual* CBR, cases take the form of words, sentences, or even full text documents. From a practical point of view, it is reasonable to apply a well-defined, restricted vocabulary set involving synonyms (cf.: [5, 6]).

A CBR system is referred to as *conversational* when the cases are represented as lists of questions and possible answers. These cases can be organized into groups, and in this way the information granules of higher level are created.

In the structural CBR approach, the *structured* set of features is called a *domain model*. Such a representation defines the maximum complexity that a case can have.

Note that the case-based classification (CBC) is a kind of CBR task. A *classifier* is a function which transforms S into K, where K denotes the number of subsets S_k identified in S, i.e. $S_k \subseteq S$, $k \in K$. Thus, K can be understood as the number of labels which can be assigned to objects in S.

For a *case-based classifier* the following notation is used

$$(Sc, sim)$$

where $Sc \subset S$ while *sim* is defined on $S \times Sc$.

Assuming that the other objects used for comparison are already labelled, the class of a new object c_i is determined using the defined form of *sim*. Usually, the nearest labelled neighbour is considered. A similar approach consists in finding k most similar cases and voting for choice of a proper neighbour. A new case in CBS is given by the description of an object (problem), with the aim of assigning the correct label (solution) to this object. In CBC, the case $c_i = (p, sol)_i$ as defined in (1) is completely determined by the problem, because the label (class) is uniquely assigned to the object (we assume that multi-label classification is out of consideration). In other words, if k is identified as the label (class) of case c_i, then $c_i \in S_k$.

If for two cases $c_i = (p, sol)_i$ and $c_j = (p, sol)_j$ their problems' descriptions p_i and p_j are similar then to both cases c_i and c_j the same class or similar classes can be assigned. However, it is necessary to define what *similarity of the classes* means.

As a result of learning, which is a key part of a CBR (CBC) system, an initially approximate classifier function can be adjusted. The learning can be performed by modification of the similarity (or distance) measure, or by supplementing the case base with new instances.

The CBR system typically involves the process of *case adaptation*. In CBC, the situation is usually much simpler. For example, if the similar case found is the nearest labelled neighbour, then its solution is the best known one, and only by performing an external validation, called *revision*, a new solution can be proposed. Sometimes this can lead to an introduction of a new label k, and consequently – the extension of the set K.

A good illustrative example is the Wisconsin Breast Cancer data [7, 8] – a popular and broadly examined dataset. This repository consists of breast cancer patients records (in terms of CBC – cases ci). Each sample is described by nine attributes (valued from 1 to 10) and an additional attribute that represents the diagnosis (two possible values – benign or malignant). The attributes describing the disease are: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli and mitoses [7]. In terms of the notions used in definition (1), the nine attributes mentioned constitute p, and the diagnosis – the solution *sol*.

Since the records in a database are not always complete (like in the Wisconsin Breast Cancer data base) sometimes the processing must rely on portions of cases.

In domains such as medicine, it makes sense to extend the case description (1) to the triple

$$c_i = (p, sol, eff)_i \tag{2}$$

where:

- p – set of observations (symptoms, images, signals, etc.);
- *sol* – diagnosis, or diagnosis with treatment;
- *eff* – prognosis.

The information to be recorded as a *case* can involve various types of data: numbers, textual records, signals, and images. For effective use, from more complex information sources like textual records, signals, and images, some representation of the relevant information needs to be extracted. For example, in the case of textual documents a sophisticated parsing is performed. In the case of images – features of detected objects are extracted. This information provides the basis for case description.

As stated above, the process of pattern recognition (PR) comprises three main mechanisms: data or information acquisition, feature selection or extraction, and classification or clustering. The natural parallel between PR and CBR leads us to implementation of the PR requirements and tasks within the CBR system. The parallel representation is as follows. Data or information acquisition is realized by creating the base of previously solved cases (*Sc*). At this stage, feature selection or extraction is performed, as well as at the moment of query formulation (description of a new case to be evaluated). Finally, the CBR cycle is performed to achieve the expected PR result.

3. Parametrized Multidimensional Clustering and Classification

3.1. Active hypercontours

First introduced in [9], the idea of active hypercontours (AH) was developed as a generalization of the traditional AC techniques. The hypercontour can be used to separate any set of objects described by features in metric space X into an arbitrarily chosen number of classes (regions) L. The formal definition is given below; see also [9, 10, 11]:

Definition 1. *Let ρ denote any metric in X, $\boldsymbol{L} = \{1, ..., L\}$ denote the set of labels and let $K(x_0, \epsilon) = \{x \in X : \rho(x_0, x) < \epsilon\}$ denote the sphere with centre $x_0 \in X$ and radius $\epsilon > 0$. The set $h \subseteq X$ with information about labels of regions it surrounds, is called a hypercontour if and only if there exists a function $f : X \to R$ and $p_0 = -\infty, p_1 \in R, ..., p_{L-1} \in R, p_L = \infty$ such that:*

$$h = \{x \in X : \exists_{l_1, l_2 \in L, l1 \neq l2} \forall_{\epsilon > 0} \exists_{x_1, x_2 \in K(x,\epsilon)} \omega(x_1, l_1) \wedge \omega(x_2, l_2)\}, \tag{3}$$

where condition $\omega(x, l)$ is true only when $p_{l-1} < f(x) < p_l$ and region $\{x \in X : \omega(x, l)\}$ represents class $l \in \boldsymbol{L}$.

As demonstrated in [9], a hypercontour is equivalent to a classifier if $X = R^n$ and $n \in N$. This statement holds for any other metric space X (the proof is almost identical). It follows from the above that each classifier generates a hypercontour in each metric space X which has a sufficient discriminative power to distinguish classified objects, and conversely, each hypercontour unambiguously generates the corresponding classification function. The term hypercontour is used to highlight the relationship of the proposed technique with active contour methods. The concept of active hypercontour defined above, although is it convenient for theoretical considerations, for practical use, however, requires a specific implementation approach. A possible solution is to use the potential active hypercontour (PAH) proposed in [12]. Its generalization for any metric space is presented below.

Definition 2. *Let feature space X be a metric space with a metric $\rho : X \times X \to R$ The potential hypercontour is defined by means of a set of labeled control points:*

$D^c = \{(\boldsymbol{p}^{\boldsymbol{c}}_{\boldsymbol{1}}, l^c_1), \ldots, (\boldsymbol{p}^{\boldsymbol{c}}_{\boldsymbol{N^c}}, l^c_{N^c})\}$ *where* $\boldsymbol{p}^{\boldsymbol{c}}_{\boldsymbol{i}} \in X$ *and* $l^c_i \in \boldsymbol{L}$ *for* $i = 1, \ldots, N^c$. *Each point is a source of potential the value of which decreases with increase of distance from the source point. The classifier k (and consequently the corresponding hypercontour h which it generates) is defined in the following way:*

$$\forall_{\boldsymbol{x} \in X} k(\boldsymbol{x}) = argmax_{l \in L} \sum_{i=1}^{N^c} P_{\Psi_i, \mu_i}(\boldsymbol{p}^{\boldsymbol{c}}_{\boldsymbol{i}}, \boldsymbol{x}) \delta(l^c_i, l), \tag{4}$$

where $\delta : L \times L \to \{0, 1\}$, $l_1 \neq l_2 \Rightarrow \delta(l_1, l_2) = 0$, $l_1 = l_2 \Rightarrow \delta(l_1, l_2) = 1$ *and* $P : X \times X \to R$ *is potential function e.g. exponential potential function:*

$$P_{1\ \Psi,\mu}(\boldsymbol{x_0}, \boldsymbol{x}) = \Psi e^{-\mu \rho^2(\boldsymbol{x_0}, \boldsymbol{x})} \tag{5}$$

or inverse potential function:

$$P_{2\ \Psi,\mu}(\boldsymbol{x_0}, \boldsymbol{x}) = \frac{\Psi}{1 + \mu \rho^2(\boldsymbol{x_0}, \boldsymbol{x})} \tag{6}$$

In both cases $\Psi \in R$ and $\mu \in R$ are parameters characterizing the potential field. Those parameters and the distribution of control points fully describe the classifier. As mentioned above, the chief advantage of the active hypercontour method is the fact that it is capable of defining *energy* (the objective function) in an almost arbitrary way. Each classifier assigns labels to vectors from the feature space and divides it into L regions of different topology. The boundaries of those regions represent visually the hypercontour. In the special case determined by $n = 2$ and $L = 2$, an image can be divided into two regions, and the boundary of the part interpreted as the object is in fact a visual representation of the contour.

In Figure 2, a sample result of the potential active hypercontour applied to the IRIS database [13] using the first and the second feature is presented. For the sake of simplicity of binary classification, one of the classes (iris setosa, crosses) was classified as the object, while the other two were assumed as the background (dashes). The energy function used in optimization is described below (8). The first study to describe the relationship between active contour methods and the classifier construction techniques was [14]. The relationship is quite obvious since both groups of methods apply the same scheme: they define a parametrized model of the object to be tracked, choose an appropriate function to evaluate the objects within that model and, finally, define the optimization method to establish an optimal parameter setting to delineate the contour that best matches the desired object outline. The present study focuses on the C^{model}, i.e. the adaptive potential active contour [15]. The adaptive potential active contour may be applied as a binary or multiclass classifier. The energy function used in contour evaluation can be chosen to fit in supervised or unsupervised learning optimization process. In feature space

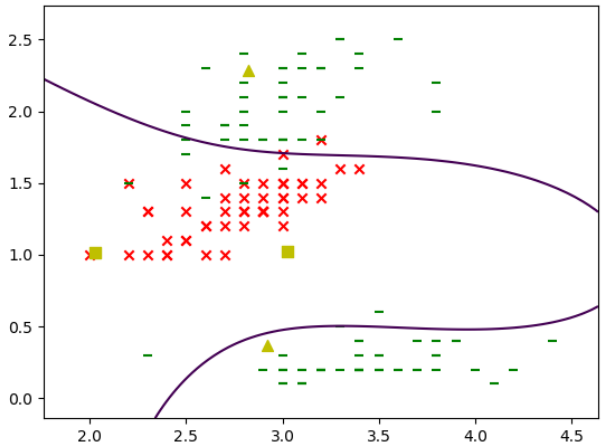

Figure 2. Sample hypercontour, L = 2 classes (the values of parameters Ψ and μ are 1.0 and 10.0, respectively)

X, it sets an arbitrary number of potential source points that define a potential field comparable to an electric field in physics. Each point is a source of potential assigned to one of labels L. In the case of a binary classifier, it divides the feature space into two subspaces, one with a positive and one with a negative potential. In Figure 2 the subspace with a positive potential (top left) is called the *object* while the rest are defined as *background*. To be more formal, classification in adaptive potential active contour method can be described as follows:

$$k(\boldsymbol{x}) = sgn[\sum_{i=1}^{N^c} P_{\Psi_i,\mu_i}(\boldsymbol{x}, \boldsymbol{p_i})], \tag{7}$$

where $\boldsymbol{p_i} \in X$ is a source of potential (potential point) in feature space X, $P : X \times X \rightarrow R$ is the potential function of distance from point $\boldsymbol{p_i}$ with additional parameters Ψ, μ and with given metric function $\rho : X \times X \rightarrow R$ in feature space X. In the performed experiments, the inverse potential function (6) and the Euclidean distance are used. The energy function used to evaluate the contour in supervised learning is commonly defined as:

$$E(k) = \sum_{i=1}^{N^c}(1 - \delta(l_i, k(\boldsymbol{x_i}))), \tag{8}$$

where k is the adaptive potential active contour being evaluated, $\boldsymbol{x_i}$ and l_i are the data point and the corresponding label, respectively, and δ is an indicator function.

The most common methods used for minimizing the energy function and, consequently, selecting the optimal solution, are the simulated annealing optimization and the genetic algorithms.

A major challenge associated with the use of potential active contours is the choice of proper values of number N^{lo} of objects sources and N^{lb} of background sources. If those numbers are too small, some of the shapes may turn out to be unreachable, e.g. if only one object source or one background source is chosen, only simple circular shapes can be found if the whole contour lies inside the image. Too large values, on the other hand, can slow down significantly the optimization process and result in the global optimum being difficult to reach as the dimension of the search space grows. One solution is to set it arbitrarily or after a series of initial experiments, which is an approach adopted in this study. An alternative method, presented in [16], consists in using the adaptation algorithm, with an initially small number of sources and additional sources introduced after each optimization phase until the satisfactory energy value is reached.

3.2. Optimisation of the contour

The contours generated by a given method (snake [17], potential [9], or others) are subject to quality assessment, which takes into account three elements (features):

(a) current contour shape features – represented and evaluated by the value of E_{int}, called *internal energy*; *attribution*;
(b) features of the background (image) and position of the contour on the image – *image energy* E_{img}; *evidence*;
(c) external knowledge or (user) demands related to the contour – *constraint energy* E_{con}; *specification*.

The general objective function E (called energy) used for the evaluation of the contour is usually of the form:

$$E = E_{int} + E_{img} + E_{con} = E_{int} + E_{ext} \tag{9}$$

where external energy $E_{ext} = E_{img} + E_{con}$. The classic internal energy is of the form:

$$E_{int} = \int_0^1 \frac{\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2}{2} ds \tag{10}$$

where:

- $s \in [0, 1]$ - position of the point on the contour,
- $v(s) = (x(s), y(s)$ - coordinates of the considered point,
- $\alpha \in [0, 1]$ - elasticity parameter, [

– $\beta \in [0, 1]$ - rigidity parameter.

Thus, the search for an optimal hypercontour is performed by optimisation of performance index E called energy

$$E : H \to R^{+} \cup \{0\} \text{ with } H \text{ being the space of all available hypercontours.}$$

As demonstrated in [9] every hypercontour generates a corresponding classification function. This holds true if the space X is metric. In E almost any type of information can be used provided that it can be implemented in a computer- oriented form, such as a mathematical formula, an output of a neural network, an output of a fuzzy system, etc. The classification can be *supervised* or *unsupervised*, with the former being more intuitive.

A hypercontour has a limited discrimination capability, which depends on the number of control points (assuming that the other parameters are fixed). The flexibility of the potential active hypercontours (PAH) can be enhanced if the optimisation procedure allows a change in the number of control points. This process is referred to as *adaptation*. A high misclassification rate in some areas of space X can be the reason for introducing a number of new control points (*local tuning* in certain regions is also possible).

The form of energy should be such that in the form of its minima the *concept* of the shape to be recognized is reflected. In other words, the external *domain expert knowledge* or *user expectation* should be implemented. Operation on elements of high granularity in a natural way leads to the use of linguistic descriptions of desired shapes, which allows one to express the domain knowledge in a way resembling a natural language. At least three approaches can be applied:

- concepts are transformed into mathematical formulae;
- knowledge is encapsulated in rules formulated in a natural language; the reasoning process is implemented in fuzzy controller based on those rules;
- graphs represent knowledge about the image and concept; vertices represent (semantically defined) objects; edges – linguistic relations between the objects detected.

The idea of using expert opinion was proposed earlier in the unsupervised classification problem where an expert estimated the proximity between pairs of objects from the training set. The search for the optimal hypercontour may be conducted in many ways, e.g. by the use of simulated annealing or genetic algorithm, both of which perform a global search without using gradient.

4. Justifiable Granularity and Hypercontour-Based Case Specification

The basic item subject to information processing in CBR was defined as follows [6]:

Definition 3. *An* information entity *(IE) is an atomic knowledge item in the domain. It represents the lowest granularity of knowledge representation.*

For instance, an IE may be represented by a particular attribute-value pair.

A *case* is considered as a set of information entities [6]. Given Def. 3, a *case* is a collection of information items of lowest granularity.

The concept of *information granule* has been widely used in recent years. Information granules are intuitively appealing constructs, which play a vital role in human cognitive and decision-making activities [3, 4, 18]. The higher the granularity (the number of elements embraced by the information granule), the higher the abstraction of this granule and the lower its specificity.

However, to develop and implement a system working with information granules we need some more precise definition.

Definition 4. *Information granule: a collection of elements (e.g. data) that are arranged in such a way as to carry the semantics relevant to the intended level of abstraction and the goal for which it has been created.*

The granularity increases as the number of such entities decreases; it is a non-increasing or decreasing continuous function of this number of entities [3, 4, 18]. Two examples:

(a) In computer image analysis, the following information granules can be considered and used for image content interpretation: pixels (basic granule of lowest level), superpixels (i.e. groups of pixels), image segments, objects detected, objects with neighbourhood creating context, whole image.
(b) In textual document analysis, the following information granules are considered: single letters, characters, syllables, words, synonyms, sentences, paraphrases, full text documents.

The construction of information granules is based on certain criteria, such as spatial neighbourhood (e.g. superpixels), similarity or functionality. For example:

- an information granule can be simply an attribute-value vector over a finite set of attributes;
- an information granule can explicitly reflect a particular structure, like relation r between two objects o_1 and o_2: (r, o_1, o_2).

The principle of *justifiable granularity* [19, 20] describes the intuitively motivated requirements that must be fulfilled for an information granule to be considered meaningful. The requirements are as follows:

(a) *Experimental evidence*. The numeric evidence accumulated within the bounds of the granule considered has to be as *high* as possible. It should reflect as big amount of data as possible to make the data set legitimate.
(b) *Well-articulated semantics*. At the same time, the granule should be as *specific* as possible. It should carry a well-defined semantics (meaning). The agreement with human-user perception of knowledge about the problem is desired.

As these two requirements are in conflict, finding a compromise is of great practical importance.

Note that information granularity can have a hierarchical structure. Two kinds of constructs were introduced: *higher-type* information granule – where the description of information granules is provided in the form of information granules themselves, and *higher-order* information granule – a universe of discourse composed of information granules. The use of such constructs can be legitimated by the practical need of solving the problem at hand. For example, it is easy to see that an information granule can be composed of information granules of lower level, which represent more detailed information, or simple data.

The area of research that deals with the representation, construction and processing of information granules is called *granular computing* [22,23]. The use of granular computing enables human-centric information processing.

In Section 2, the case has been defined as an ordered pair $c = (p, sol)$ with p and sol representing the problem's description and solution, respectively.

We assume that both p and sol are composed of information granules or constitute by themselves information granules that fulfil the requirements of CBR justifiable granularity. For example, the information granule p (or sol) includes a number of elements (also granules of lower type) that are justified as legitimate descriptors of the problem under consideration.

Note that the number of elements in p (or sol) forms a simple descriptor of information granularity. Moreover, the number of information granules in a case may be variable. For example, to find the correct solution of a new case, only a part of the information available may be sufficient. However, as it often happens in the medical domain, new information may be required if the currently available data do not allow a physician to make a definite diagnosis.

According to Def. 2 given in Section 3, the potential hypercontour is defined by a set of labelled control points x_i^c and parameters of the applied potential functions P_i^c characterizing the potential field where each point x_i^c is a source of potential P_i^c. In other words, those parameters as well as the distribution and number of control points (*adaptation* and *local tuning*) fully describe the hypercontour.

The combination of the CBR and the hypercontour approach requires proper definition of the case $c_i = (p, sol)_i$, cf. (1). This involves the following elements:

1. labelled control points x_i^c;
2. determined parameters of the source of potential P_i^c;
3. number of control points N^c.

Working with the imposed constant number of control points, one can simply state that

$$c_i = ([x_i^c, P_i^c]^T, sol)_i \tag{11}$$

where *sol* is the result of recognition within all the admissible potential sets $\vartheta(i)$ representing possible recognition results.

A crucial task of the CBR system is to determine the similarity between query *new* $[x_i^c, P_i^c]^T$ and each of the *old* cases collected in *Sc* (cf. Fig. 1.) Thus, the similarity measure must be defined properly. The definition (11) enables the application of well-known standard vector similarity measures.

The case base *Sc* collects cases (hypercontours) obtained by diverse number of control points N^c. The cases have diverse representations because of diverse dimensions of vectors x_i^c. This situation requires the application of non-standard measures and approaches. The possible solutions include:

(a) Reduction of the dimension of one of the vectors. However, the problem is to motivate deeply the decision about the removed elements.
(b) Application of correlation measure defined for vectors of diverse dimensions [21]. However, this solution lacks a persuasive interpretation concerning the shape of the hypercontour and, consequently, about the classification label.

Fortunately, one has the final value of the energy (9) obtained after the optimisation has been performed. The value of E can be alternatively used as a decision factor if the compared cases are "similar".

The active and adaptive hypercontours approach can be formally interpreted using the following statements:

- Each data element (in images: pixel, line segment, superpixel, etc.) can be described by its features (for pixel – coordinates and colour components).
- Hypercontour acting on the elements of j-th level of granularity ($j = 0, 1, 2, \ldots$) determines potential sets $\vartheta(i)$ representing possible recognition results.

- Each set $\vartheta(i)$ is evaluated by the energy. As a result, the energy function can be regarded as a concept recogniser.
- The optimisation of the energy function leads to making the decision. Thus, one may speak of decision function d that is able to find a proper set $\vartheta(i)$.
- The energy function can model any arbitrary concepts and relations. The high-level knowledge is usually obtained from domain experts or by the use of a machine learning approach.

In the case of graph representation, the optimal prototype matching is performed [22].

The existence and successful applicability of recognition methods on diverse levels of information granularity is of key importance for performing complex data analysis. Nevertheless, all the elements should be composed wisely, i.e. the way of justifiable granulation, representation of granules and their relations, representation of concepts, consideration of context, detection and treatment of outliers, and recognition method. These remarks apply to all kinds of data.

5. Summary

The granular approach is related to intelligent analysis of all kinds of data in each metric space which has a sufficient discriminative power to distinguish classified objects. It has been shown that active hypercontours can work on the level of information granules and they can be used for the purpose of intelligent and task-oriented parametrized representation of multidimensional data. The paper has demonstrated that active contours can be incorporated into the case-based reasoning and case-based classification processes. In this way, a novel pattern recognition method has been developed. The study has also proposed ways to handle diverse representations of collected cases. The universality of the method allows it to be applied in any multidimensional metric space where the hypercontour can be used to separate any set of objects described by features in this space into an arbitrarily chosen number of classes (regions).

Acknowledgment

The author would like to express his gratitude to Arkadiusz Tomczyk, PhD (Lodz University of Technology, Poland) for his effort in developing the potential active contour approach and formal definitions of active contours, and Łukasz Pierścieniewski, MSc (graduate of Lodz University of Technology) for his help in providing the computational results.

References

[1] Pradipta Maji, S. K. P. *Rough-Fuzzy Pattern Recognition*. Wiley-IEEE Computer Society Pr, 2012. ISBN 111800440X.

[2] M Narasimha Murty, V. S. D. *Introduction to Pattern Recognition and Machine Learning*. WSPC, 2015. ISBN 9814335452.

[3] Andrzej Bargiela, W. P. *Granular Computing*. Springer US, 2012. ISBN 9781461510338.

[4] Bargiela, A. and Pedrycz, W. Toward a theory of granular computing for human-centered information processing. *IEEE Transactions on Fuzzy Systems*, 16(2):320–330, 2008. doi:10.1109/tfuzz.2007.905912.

[5] Montani, S. and Jain, L. C., editors. *Successful Case-based Reasoning Applications-1*. Springer Berlin Heidelberg, 2010. doi:10.1007/978-3-642-38736-4.

[6] Lenz, M., Burkhard, H.-D., Bartsch-Spörl, B., and Wess, S., editors. *Case-Based Reasoning Technology*. Springer Berlin Heidelberg, 1998. doi: 10.1007/3-540-69351-3.

[7] Street, W. N., Wolberg, W. H., and Mangasarian, O. L. Nuclear feature extraction for breast tumor diagnosis. In R. S. Acharya and D. B. Goldgof, editors, *Biomedical Image Processing and Biomedical Visualization*. SPIE, 1993. doi:10.1117/12.148698.

[8] Wolberg, W. H., Street, W., and Mangasarian, O. Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. *Cancer Letters*, 77(2-3):163–171, 1994. doi: 10.1016/0304-3835(94)90099-x.

[9] Tomczyk, A. Active hypercontours and contextual classification. In *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*. IEEE, 2005. doi:10.1109/isda.2005.16.

[10] Szczepaniak, P. S. and Pierścieniewski, u. Rule extraction from active contour classifiers. *Journal of Applied Computer Science*, page Tom 26 Nr 1 (2018): Journal of Applied Computer Science, 2019. doi: 10.34658/JACS.2018.26.1.73-86.

[11] Tomczyk, A. Image segmentation using adaptive potential active contours. In *Advances in Soft Computing*, pages 148–155. Springer Berlin Heidelberg, 2007. doi:10.1007/978-3-540-75175-5_19.

[12] Tomczyk, A. and Szczepaniak, P. S. Adaptive potential active hypercontours. In *Artificial Intelligence and Soft Computing – ICAISC 2006*, pages 692–701. Springer Berlin Heidelberg, 2006. doi:10.1007/11785231_72.

[13] Dua, D. and Karra Taniskidou, E. Iris dataset. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

[14] Tomczyk, A. and Szczepaniak, P. S. On the relationship between active contours and contextual classification. In *Advances in Soft Computing*, pages 303–310. Springer Berlin Heidelberg, 2005. doi:10.1007/3-540-32390-2_35.

[15] Tomczyk, A. and Szczepaniak, P. S. Adaptive potential active hypercontours. In *Artificial Intelligence and Soft Computing – ICAISC 2006*, pages 692–701. Springer Berlin Heidelberg, 2011. doi:10.1007/11785231_72.

[16] Szczepaniak, P. S., Tomczyk, A., and Pryczek, M. Supervised web document classification using discrete transforms, active hypercontours and expert knowledge. In *Web Intelligence Meets Brain Informatics*, pages 305–323. Springer Berlin Heidelberg, 2006. doi:10.1007/978-3-540-77028-2_18.

[17] Kass, M., Witkin, A., and Terzopoulos, D. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988. doi: 10.1007/bf00133570.

[18] Pedrycz, W. *Knowledge-Based Clustering*. John Wiley & Sons, 2005. ISBN 9780471708599.

[19] Pedrycz, W. and Homenda, W. Building the fundamentals of granular computing: A principle of justifiable granularity. *Applied Soft Computing*, 13(10):4209–4218, 2013. doi:10.1016/j.asoc.2013.06.017.

[20] Pedrycz, W., Al-Hmouz, R., Morfeq, A., and Balamash, A. The design of free structure granular mappings: The use of the principle of justifiable granularity. *IEEE Transactions on Cybernetics*, 43(6):2105–2113, 2013. doi: 10.1109/tcyb.2013.2240384.

[21] Tatar, J. Miary zależności wektorów losowych o różnych wymiarac. *Zeszyty Naukowe Uniwersytetu Ekonomicznego w Krakowie*, (780):53–60, 2008. In Polish.

[22] Tomczyk, A., Pryczek, M., Walczak, S., Jojczyk, K., and Szczepaniak, P. S. Spatch based active partitions with linguistically formulated energy. *Journal of Applied Computer Science*, 18(1):87–115, 2010.