IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 3, 2017; accepted January 18, 2017, date of publication January 20, 2017, date of current version March 13, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2656618

Cooperative Profit Random Forests With Application in Ocean Front Recognition

JIANYUAN SUN, GUOQIANG ZHONG, (Member, IEEE), JUNYU DONG, (Member, IEEE), HINA SAEEDA, AND QIN ZHANG

Department of Computer Science and Technology, Ocean University of China, Qingdao 266100, China.

Corresponding author: Junyu Dong (dongjunyu@ouc.edu.cn)

This work was supported in part by National Natural Science Foundation of China under Grant 61271405 and Grant 61403353, in part by the International Science and Technology Cooperation Program of China under Grant 2014DFA10410, in part by the Open Project Program of the National Laboratory of Pattern Recognition, and in part the Fundamental Research Funds for the Central Universities of China.

ABSTRACT Random Forests are powerful classification and regression tools that are commonly applied in machine learning and image processing. In the majority of random classification forests algorithms, the Gini index and the information gain ratio are commonly used for node splitting. However, these two kinds of node-split methods may pay less attention to the intrinsic structure of the attribute variables and fail to find attributes with strong discriminate ability as a group yet weak as individuals. In this paper, we propose an innovative method for splitting the tree nodes based on the cooperative game theory, from which some attributes with good discriminate ability as a group can be learned. This new random forests algorithm is called Cooperative Profit Random Forests (CPRF). Experimental comparisons with several other existing random classification forests algorithms are carried out on several real-world data sets, including remote sensing images. The results show that CPRF outperforms other existing Random Forests algorithms in most cases. In particular, CPRF achieves promising results in ocean front recognition.

INDEX TERMS Random Forests, cooperative game theory, Banzhaf power index.

I. INTRODUCTION

Random Forests have been widely used in applications of image processing [1], [2], computer vision [3] and pattern recognition [4], which have promoted the state-of-the-art in performance. Random Forests are a type of ensemble method [5] that is mainly based on the combination of several independent decision trees [6], [7]. The original Random Forests [8] algorithm is built by combining several classification and regression trees (CART) [6]or C4.5 [7]. The essence of building decision trees is to recursively partition the sample space by using some node-split methods. Random forests have good generalization ability and do not incur over-fitting.

Although there are many versions of the random classification forests algorithms, the majority of them are using the traditional node-split method (Gini index [6] or information gain ratio [7]), such as the Rotation Random Forests algorithm [9], the random forests algorithm of Krishna et al. [10] and the random forests algorithm with ensemble of feature spaces [11], etc. However, the Gini index or the information gain ratio tends to choose an attribute that has a strong prediction power in term of the target class, which often pays less attention to the intrinsic structure of the attribute variables and fails to find attributes with good discriminate ability as a group yet weak as individuals [12]. At each node of the decision tree, each attribute variable is analyzed in turn to measure its predictive power in terms of the class output. Any combination of predicting attribute variables which presents a much stronger prediction may therefore be missed, if the operators available to the induction process are insufficient to identify that combination [13]. To cope with this demand, we propose a novel random classification forests algorithm based on the cooperative game theory. Cooperative game involves a set of players and a reward associated with different groups or coalitions of players. The reward of a certain coalition depends on individual contributions of players composing this coalition to the game. The larger the contribution of a player is, the higher benefit to have this player in a coalition. Coalitions with high reward are naturally preferable over those with small reward. In this perspective, cooperative game theory is already widely used in image processing [14], [15], complex network [16], [17] and feature selection area [18]–[20], which have promoted the state-of-the-art in performance.

Inspired by the way of building decision trees and the cooperative game theory, we propose a new random classification forests algorithm called Cooperative Profit Random Forests (CPRF). CPRF is composed of a number of new decision trees. These decision trees, called Profit Decision Trees (PDT), adopt the new node-split method, and use the *efficient* property of Banzhaf power index to calculate the best split point at each tree node. The Banzhaf power index is a method of calculating solution of the cooperative game. It can not only give the importance ranking of each attribute variable, but also 'consider' the relationship between attribute variables in the process of calculating the importance of each attribute variable (for details, see Section 3). Exploiting the *efficient* property of Banzhaf power index, we can obtain the gains (payoff) of each attribute variable of two descendants sample space. The best split point is to make the two descendants sample space have the highest gains, that is, this splitting way makes each attribute variable in the two descendants sample space obtain the reasonable profit. Using this splitting node method on the non-leaf node until the stop rule is met. Then, the attribute variable of each non-leaf node in a single tree is ''willing'' to cooperate together to construct a decision tree. We evaluate the proposed algorithm on several public available datasets by comparing it with four existing random forests methods to show its effectiveness. Experimental results show that CPRF is effective on most real-world datasets.

In addition, we try to use the different random forest algorithms to solve the challenging oceanography problem of the ocean front recognition. To our knowledge, it is the first time to use random forests on ocean front recognition problem. An ocean front is a boundary between two distinct water masses. The water masses are defined by moving in different directions, i.e., on one side of the ocean front the water is generally moving in one way, and on the other side of the ocean front, the water is moving in another. The water masses on either side of an ocean front may also have different temperatures, salinities, or densities, along with differences in other oceanographic markers [21]. Ocean front has important application significance in marine fishery, environmental protection, marine military affairs, etc. The recognition and detection of ocean front have become the important study topics in Physical Oceanography and marine discipline. According to the experimental result of the ocean front recognition, it has shown that CPRF outperformed other existing random forests algorithms.

The organization of the paper is as follows. In Section 2, we provide a brief overview of the existing random classification forests algorithms. In Section 3, we introduce the basic concept of cooperative game theory. Section 4 provides a detailed description of the Profit Decision Tree (PDT) and Cooperative Profit Random Forests (CPRF). Section 5 presents the experimental results, which obtained by CPRF and the compared algorithms on several practical datasets. In particular, remote sensing images data. Section 6 concludes the paper and proposed research prospects.

II. RELATED WORK

The original random forests algorithm [8], which combined several decision trees [6] using bagging [22], was introduced by Breiman. As is well known, Breiman's random forests mainly refer to three works: the random subspace method of Ho et al. [23], the feature selection work of Amit et al. [24] and the method of random split selection of Dietterich et al. [25]. The core ideas of decision trees [26] are also used in random forests models. The randomness of random forests mainly reflected in the number of training samples for building each tree and the number of attribute variables for each tree node. With these two guarantees of randomness, random forests algorithm will not incur over-fitting.

In recent years, many researchers have devoted effort to improve the performance of the random classification forests algorithm, so there are many versions of random classification forests algorithm. For example, Rotation Random Forests algorithm [9] has achieved higher classification accuracy than the original random forests algorithm [8]. Rotation Random Forests aim at building individual decision trees with high accuracy and diversity. The main idea of Rotation Random Forests is to apply transformation method (PCA) [27] to transform the data at each node to another space when computing the best split at this node [9]. Recently, Zhang et al. propose a new random forests model based on the idea of Rotation Random Forests [11]. This random forests algorithm [11] mainly apply two kinds of transformation methods (PCA) [27] and (LDA) [28] to transform the data at each node to another space when computing the best split at that node. In addition, Krishna et al. [10] employ the different attribute selection methods for the original random forests algorithm, which can eliminate irrelevant sample feature variables from the raw dataset. By this way, the performance of the original random forests algorithm has been improved. Besides these, Bernard et al. [29] introduce a dynamic random forests algorithm to improve the performance of random forests algorithms, which is based on a sequential procedure that builds an ensemble of random trees by making each of them dependent on the previous ones. This is done through a re-sampling of the training data, inspired by boosting algorithms, and combined with other randomization processes used in traditional random forests algorithm. Remarkably, inspired by the Boosting algorithm, Schulter et al. [30] propose a new method termed Alternating Decision Forests that extends the original random forests [8] by minimizing any given global loss function. This algorithm achieved the minimum of the global loss by updating the sample weight during training. To let the splitting functions consider the sample weights, Schulter et al. changed the standard entropy calculation to a weight entropy by changing the estimation of the class distribution.

These works mainly focus on processing sample data and selecting the excellent random trees classifier for the original random classification forests algorithm. They also employ the traditional node-split method to achieve the growth of

each tree. Recently, to effectively handing of dynamically growing datasets, two variants of random forests [4] are introduced. One is based on binary classifier SVMs, and the another on the nearest class mean classifier (NCM). Both random forests variants employ the SVMs and NCM as the node-split functions, respectively. They achieved good performance for the task of learning incrementally new classes. In addition, Sun et al. [31] propose a new node-split method of constructing decision tree for the random classification forests algorithm, and they first use a method of calculating solution of the cooperative game, Banzhaf power index, to evaluate the best attribute variable for each tree node. And then, the midpoint value of the best attribute variable is selected as the split point at each tree node. Although Sun et al. [31] propose a new attribute evaluate method for each tree node, they do not give the node-split method to search the ''optimal'' split point. In fact, although Sun et al. employ the concept of the cooperative game, they don't achieve the real cooperation between tree nodes.

In this paper, we propose an innovative random classification forests algorithm based on the cooperative game theory. It also uses the Banzhaf power index, however, we use the *efficient* property of Banzhaf power index to compute the ''optimal'' split point of the sample attributes for each tree node. Our approach can make tree nodes achieve the real cooperation between tree nodes. According to the comparative experiments on several real-world datasets including remote sensing images, it has proved that the proposed random classification forests algorithm is more effective than other existing random forests.

III. BASIC CONCEPT OF COOPERATIVE GAME THEORY

Cooperative game theory assumes that groups of players, called coalitions, are the primary units of decision-making and may enforce cooperative behavior. Consequently, cooperative games can be seen as a competition between coalitions of players, rather than between individual players. The basic assumption in cooperative game theory is that the grand coalition, the group consisting of all players, will form. One of the main research questions in cooperative game theory is how to allocate in some fairways the payoff of the grand coalition among the players [32].

In general, a cooperative game model is determined by three factors, namely, the players, the strategy(coalition) set and the gains (payoff) function.

- The players: Each player has decision-making in a cooperative game. Notation $i(i = 1, 2, \ldots, n)$ is used to represent each player, and notation $\mathcal{N}$ represents the collectives of all the participants in the game, i.e. $\mathcal{N} = \{1, 2, \ldots, n\}$. The number of players in a game is at least two players or a group with a common goal to get the biggest gains (profit), such as teams, enterprise, etc.
- The strategy (coalition) set: Each player $i(i = 1, 2, \ldots, n)$ can choose different feasible policies. However, in order to obtain its own maximum benefit, it is necessary to find a feasible policy which is feasible from beginning to end. Then, this feasible policy becomes the strategy set of I. In our paper, notation $\mathcal{S} \subseteq \mathcal{N}$ is used to represent the strategy of players.
- The gains (payoff) function: The results obtained by a game are called gains. The gains of each player in the game are not only related to the strategy chosen by his own, but also related to the strategy selected by all the players in a game. In general, a player in a game obtains the gains more than that of by himself or he participate in other coalitions, and the player is more willing to participating in this game. More total gains (the sum of gains from each player in a game) can be obtained in an optimal game. In our paper, $r(\mathcal{N})$ represents the total gains of a game, $r(i)(i \in \mathcal{N})$ represents the gains of each player. Notation $r(\mathcal{S})(\mathcal{S} \subseteq \mathcal{N})$ is used to represent the gains of a coalition set $\mathcal{S} \subseteq \mathcal{N}$.

According to the literature [32], the definition of cooperative game can be expressed as follows.

A cooperative game $\Gamma = (\mathcal{N}, \gamma)$ consists of a player set $\mathcal{N} = \{1, 2, \ldots, n\}$ and a characteristic function $\gamma : 2^{\mathcal{N}} \to R$. For each subset $\mathcal{S} \subseteq \mathcal{N}$, $\gamma(\mathcal{S})$ can be interpreted as the profit achieved by the players in $\mathcal{S} \subseteq \mathcal{N}$. One of the main research questions in cooperative game theory is how to allocate the total gains $\gamma(\mathcal{N})$ for each player that belongs to the grand player coalition $\mathcal{N}$ (i.e., each player $i \in \mathcal{N}, i = 1, 2, \ldots, n$) in a fair and reasonable way.

Different solution concepts based on different notions of fairness and rationality have been proposed in the cooperative game theory literature, such as the ''core'', the ''Banzhaf power index'' and some concepts related to approximate solutions. Among these concepts, the Banzhaf power index is motivated by fairness, which can produce a unique solution for each player in a game.

The original definition of the Banzhaf power index is described in [33]. In a simple game, $\Gamma = (\mathcal{N}, \gamma)$ consists of a player set $\mathcal{N} = \{1, 2, \ldots, n\}$ with $|\mathcal{N}| = n$. The coalition $\mathcal{S}$ with value 1 are called 'winning', and that with value 0 is called 'losing', i.e. $\gamma(\mathcal{S}) = 1$ and $\gamma(\mathcal{S}) = 0$, respectively. The phenomenon that coalition $\mathcal{S} \cup \{i\}$ wins but $\mathcal{S}$ loses is called a swing of player $i \in \mathcal{N}$, because the membership of player i in the coalition $\mathcal{S} \cup \{i\}$ is crucial to its 'winning'. The Banzhaf power index of a player $i \in \mathcal{N}$ is the probability of swings of player i. We denote the Banzhaf power index of player $i \in \mathcal{N}$ as $\beta_i(\Gamma)$ and it is given by

$$\beta_i(\Gamma) = \frac{1}{2^{n-1}} \sum_{\mathcal{S} \subseteq \mathcal{N} \setminus \{i\}} \Delta_i(\mathcal{S}), \tag{1}$$

where 2^{n-1} is the total number of subsets $\mathcal{S} \subseteq \mathcal{N} \setminus \{i\}$, $\Delta_i(\mathcal{S})$ is the marginal contribution of player i. i.e. $\Delta_i(\mathcal{S}) = \gamma(\mathcal{S} \cup \{i\}) - \gamma(\mathcal{S})$.

According to Eq. (1), the Banzhaf power index of player $i(i \in \mathcal{N})$ is mainly to count the number of 'winning' coalitions, when the player i joins in the 'losing' coalitions $\mathcal{S} \subseteq \mathcal{N} \setminus \{i\}$. The normalized Banzhaf power index $\eta_i(\Gamma)$ is defined

as

$$\eta_i(\Gamma) = \frac{\beta_i(\Gamma)}{\sum_{i\in\mathcal{N}} \beta_i(\Gamma)}, \tag{2}$$

where $\sum_{i\in\mathcal{N}} \beta_i(\Gamma)$ can be considered as the total gains of all the 'winning' coalitions in a game. Obviously, a player with the maximum value of Banzhaf power index $\eta_i(\Gamma)$ has played an important role for the final results of a game.

Banzhaf power index has a particularly interpretation — it measures the power of the player in a game, i.e., the probability that this player can bring good or bad results to a game. Banzhaf power index has been widely applied to solve some practical problems, such as, evaluating the distribution of power in UN Security Council. Some famous examples in the literature [34] can help reader better understand the application and calculation of Banzhaf power index. The Banzhaf power index value $\eta_i(\Gamma)$ has many attractive properties. In our paper, the *efficient* property of the Banzhaf power index, i.e. Proposition 1 [32], is used in our decision tree algorithm.

Proposition 1: For any characteristic function game $\Gamma = (\mathcal{N}, \gamma)$, we have $\sum_{i=1}^{n} \eta_i(\Gamma) = \gamma(N)$, i.e., The sum of the Banzhaf power index of the individual player $i(i = 1, 2, \ldots, n)$ is equal to the total gains of the game $\Gamma = (\mathcal{N}, \gamma)$.

In our paper, each step of building decision trees in the random forests can be considered as a cooperative game $\Gamma = (\mathcal{N}, \gamma)$ where the attribute variables are the players in this game, i.e., the grand player set $\mathcal{N} = \{f_1, f_2, \ldots, f_n\}$ and the single decision tree also employs an exhaustive way to search the unique split points at each tree node. We will use the *efficient* property of the Banzhaf power index to search the best split points, which makes the formation of the two descendants sample space have the highest gains (payoff).

Here, we will explain how to calculate the Banzhaf power index Eq. (1) for each individual attribute player by referring to the method of Sun et al. [31]. The calculation procedure of Banzhaf power index is given as follows.

Let the coalition $\mathcal{S} \subseteq \mathcal{N}$ be subset of the attribute player set and $f_i \in \mathcal{N}(f_i \notin \mathcal{S})$ be a attribute player to be estimated. Recall the definition of the Banzhaf power index and Eq. (1), and the Banzhaf power index is used to count number of 'winning' coalition when the player i joins in the 'losing' coalitions $\mathcal{S} \subseteq \mathcal{N} \setminus \{i\}$. Therefore, we define the ratio $p = \mu_{f_i}(\mathcal{S})/\rho_{f_i}(\mathcal{S})$ to measure whether the attribute player f_i can make a coalition $\mathcal{S}$ enter the 'winning' state. Where $\mu_{f_i}(\mathcal{S})$ is the number of attributes (belonging to the coalition S) interdependent with the attributes $f_i(f_i \notin \mathcal{S})$, and $\rho_{f_i}(\mathcal{S})$ is the total number of attributes in the coalition $\mathcal{S}$. For convenience, we define a threshold value τ for the ratio p and set $\tau = 1/2$. If $p < \tau$, we consider the coalition $\mathcal{S} \cup \{f_i\}$ to be 'losing', otherwise, it is 'winning', i.e.,

$$\Delta_{f_i}(\mathcal{S} \cup f_i) = \begin{cases} 1 & p \geq \tau; \\ 0 & p < \tau. \end{cases} \tag{3}$$

The threshold $\tau = 1/2$ means, if more than half of the attributes of a coalition $\mathcal{S}$ are interdependent with f_i, then f_i joining coalition $\mathcal{S}$ can make it enter the 'winning' state. We use the conditional mutual information [35] to evaluate the interdependence between a single attribute $f_i \in \mathcal{N} \setminus \mathcal{S}$ and the attribute player $f_j \in \mathcal{S} \subseteq \mathcal{N}$.

Here, the conditional mutual information is approximated as the amount of interdependency between a single attribute player $f_j \in \mathcal{S}$ and attribute player $f_i \notin \mathcal{S}$. $I(f_j; \mathbf{y}|f_i)$ is used to evaluate the impact of new emerging variable $f_i \notin \mathcal{S}$ on the information shared by variables $f_j \in \mathcal{S}$ with the class $\mathbf{y}$. It is defined by

$$I(f_j; \mathbf{y}|f_i) = p(f_j, \mathbf{y}, f_i) \log \frac{p(f_j, \mathbf{y}|f_i)}{p(f_j|f_i)p(\mathbf{y}|f_i)}, \tag{4}$$

where $\mathbf{y}$ represents the target class of the data sample. Two attribute variables f_j and f_i are independent on each other if the relevant between f_j and target class can be increased conditioned by f_i, i.e.,$I(f_j; \mathbf{y}|f_i) > I(f_j; \mathbf{y})$, where $I(f_j; \mathbf{y}) = p(f_j, \mathbf{y}) \log \frac{p(f_j, \mathbf{y})}{p(f_j)p(\mathbf{y})}$ represents the mutual information between the attribute variable f_i and the target class $\mathbf{y}$. Mutual information is widely used to measure the relevance between variables.

In addition, by Eq. (3), we can obtain simplicity of the computation for a single coalition $\mathcal{S} \subseteq \mathcal{N} \setminus \{f_i\}$ in Eq. (1), i.e.,

$$\Delta_i(\mathcal{S}) = \gamma(\mathcal{S} \cup \{i\}) - \gamma(\mathcal{S}) = \begin{cases} 1 & p \geq \tau; \\ 0 & p < \tau. \end{cases} \tag{5}$$

By Eqs. (1), (5), (2) and (4), we can obtain the Banzhaf power index of each attribute variable.

In addition, according to the literature [31], calculating the Banzhaf power index of each attribute requires summing over all possible attribute subsets, which may lead to high computational complexity. But, empirically, it is unnecessary to consider large-scale attribute subsets. Hence, we also set a bound for the size of attribute subsets, i.e. Eq. (1) can be redefined as

$$\beta_i(\Gamma) = \frac{1}{|\Pi_\varpi|} \sum_{\mathcal{S}\subseteq\Pi_\varpi} \Delta_i(\mathcal{S}),$$

where Π_ϖ is the subset of the attribute set $F \setminus \{f_i\}$ with a number of elements less than or equal to ϖ. In our paper, 5-fold cross is used to select the parameter ϖ. The experimental results show that our algorithm performs better when the value $\varpi \in [3, 5]$ for the majority of datasets. Thus, we suggest the value $\varpi \in [3, 5]$ being a bound on the subset size in practical application.

IV. ALGORITHM

A. PROFIT DECISION TREE

The Profit Decision Trees (PDT) used in the Cooperative Profit Random Forests (CPRF) are based on the binary recursive partitioning trees. These trees partition the data sample space by using a sequence of binary partitions (splits) on

individual variables. The "root" node of the tree comprises the entire sample space R^n. The tree nodes that are not split are called "leaf nodes" and form the final partition of the data sample space. Each non-leaf node splits into two descendant nodes, one on the left and one on the right, according to the split-point value of a candidate attribute variable.

In PDT, the "best" split point of a tree node can make the two descendants obtain the maximum gains (payoff). In order to find the best split point, we consider every possible split point on each candidate attribute variable and choose the best split point according to the maximum gains (payoff) criterion. In the following, we give this criterion in detail.

Given n labeled samples $D = \{(x_1, y_1), \ldots, (x_n, y_n)\} \in R^{n\times(p+1)}$ as training data, with attribute variables $f_j = (x_{1,j}, \ldots, x_{n,j})^T, (j = 1, \ldots, p)$. For every non-leaf node k, we associate a block $B_k \subseteq R^{n\times(p+1)}$ of the input sample space as follows: let $B_{root} := D$. Each internal node associated with a split $(f_j, \epsilon_{i,j})$ $(i = 1, \ldots, n-1)$, where $f_j = (x_{1,j}, \ldots, x_{n,j})^T$ $(j \in (1, \ldots, p))$ denotes the split attribute variable and $\epsilon_{i,j}$ $(i = 1, \ldots, n-1)$ denotes the split location along the attribute variable f_j $(j \in (1, \ldots, p))$, the value of the split point is $\epsilon_{i,j} = (x_{i,j} + x_{i,j+1})/2$, $i = 1, \ldots, n-1; j \in (1, \ldots, p)$. We then have:

$$B_{left(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} \le \epsilon_{i,j}\};$$
$$B_{right(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} > \epsilon_{i,j}\}.$$

According to the current split point $\epsilon_{i,j} = (x_{i,j} + x_{i,j+1})/2$, the tree produces the two candidate descendants as $B_{left(f_j)}$ and $B_{right(f_j)}$. Define the notation NL and NR that are used to denote the available attribute variables dataset of $B_{left(f_j)}$ and $B_{right(f_j)}$, respectively. Then, we compute the Banzhaf power index for the individual attribute variable in the sample space NL and NR. By the *efficient* property of the Banzhaf power index, i.e. Proposition 1:

$$\sum_{j=1}^{p} \eta(f_j) = \gamma(N), N = (f_1, \ldots, f_p) \in B_{left(f_j)} or B_{right(f_j)}.$$

The best split point is chosen to the maximum value of $split(f_j, \epsilon_{i,j}) = \gamma(NL) + \gamma(NR), i = 1, \ldots, n-1; j \in (1, \ldots, p)$.

We will use this method recursively to select the best split point in a decision tree, until a stop rule is met. In our paper, the node-split procedure will stop when a split node contains fewer samples. Then, we use notation T to represent a single Profit decision tree (PDT). The generative process of PDT is described in the following, i.e. Algorithm 1.

B. COOPERATIVE PROFIT RANDOM FORESTS ALGORITHM

CPRF employs the general technique of bootstrap aggregating, or bagging, to the PDT learner. The main algorithm flow of CPRF is described as Algorithm 2.

In our paper, in order to efficiently deal with large scale datasets, we divide the samples D_l into two parts by using the information gain rate to search the best split point at the root node of the individual PDT in the CPRF. Then, for the two descendant nodes of the root node and the other nodes except the leaf nodes, the new node-split method is used.

Algorithm 1 Profit Decision Tree (PDT)

1: **Initialize**: $D = \{(x_1, y_1), \ldots, (x_n, y_n)\} \in R^{n\times(p+1)}$, as training data with attribute variables $f_j = (x_{1,j}, \ldots, x_{n,j})^T, j = (1, \ldots, p), T = \varnothing, \epsilon = 0$, let $B_{root} := D$;
2: TreeBlock(root, B_{root})

Algorithm 1.1 TreeBlock (f_j, B_{f_j})

1: Add f_j to $T, j \in (1, \ldots, p)$
While
2: Set the split point of the attribute variable f_j is $\epsilon_{i,j}$,

$$\epsilon_{i,j} = (x_{i,j} + x_{i,j+1})/2, i = 1, \ldots, n;\ j \in (1, \ldots, p-1)$$

3: For the split point $\epsilon_{i,j}$ of each attribute $f_j (j = 1, \ldots, p)$ do
4: Compute: $\gamma(NL) = \sum_{j=1}^{p} \eta(f_j), f_j \in B_{left(f_j)}$;

$$\gamma(NR) = \sum_{j=1}^{p} \eta(f_j), f_j \in B_{right(f_j)},$$

where $B_{left(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} \le \epsilon_{i,j}\}$ and $B_{right(f_j)} = \{(f_1, \ldots, f_p) \in B_{f_j} : x_{i,j} > \epsilon_{i,j}\}$.
5: $Split(\hat{f}_j, \hat{\epsilon_{i,j}}) \leftarrow \arg\max(\gamma(NL) + \gamma(NR))$, set $B_{left(\hat{f}_j)} = \{(f_1, \ldots, f_p) \in B_{\hat{f}_j} : x_{i,j} \le \hat{\epsilon}_{i,j}\}$ and $B_{right(\hat{f}_j)} = \{(f_1, \ldots, f_p) \in B_{\hat{f}_j} : x_{i,j} > \hat{\epsilon}_{i,j}\}$
6: TreeBlock(left($\hat{f}_j$), $B_{left(\hat{f}_j)}$)
7: TreeBlock(right($\hat{f}_j$), $B_{right(\hat{f}_j)}$) **Until** reaching the user-set limit, i.e., a minimal number
8: of samples of a node. 8: f_j to be the $leaves(T)$

V. EXPERIMENTS

In this section, in order to verify the performance of CPRF, the classification experiment test and the robustness analysis are carried out; we empirically compare CPRF to the original Random Forests (RF) [8], the Rotation Random Forests (ROF) [9], the Alternating Decision Forests (ADF) [30] and the Banzhaf Random Forests (BRF) [31] on several real-world datasets. In addition to test our proposed algorithm using publication available image datasets, we also evaluate our CPRF using data from medical and Physical sciences. Beyond these, we have a strong interest to evaluate the performance of the different random forests algorithms on oceanographic data from remote sensing.

A. CLASSIFICATION PERFORMANCE OF CPRF

We will briefly describe five different random forests algorithms and discuss some of the differences between them. These random forests algorithms, including the original Random Forests (RF) [8], the Rotation random forests (ROF) [9], the Alternating Decision Forests (ADF) [30], the Banzhaf Random Forests (BRF) [31] and the Coopera-

Algorithm 2 Cooperative Profit Random Forests (CPRF)

Given n labeled samples $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ as training data, with each attribute variable $f_j = (x_{1,j}, \ldots, x_{n,j})^T, j = 1, \ldots, p$.

For $l = 1, \ldots, L$,

1. Take a bootstrap samples D_l of size n from D with replacement; Select $m(m \ll p)$ attribute variables at random from the p available attribute variables.
2. Using the bootstrap samples D_l as the training data fits a tree using binary recursive partitioning (see Algorithm 1):
   a. Start with all observations in a single tree node.
      i. Find the best binary splitting points $\hat{\epsilon}_{i,j}$ among all binary splits on the m attribute variables.
      ii. Split the node into two descendant tree nodes using the split from step i.
      iii. Repeat the step i and ii until reaching the user-set limit, i.e., a minimal number of samples at a node.
3. To make a classification prediction at a test sample x,

$$\hat{f}(x) = \arg\max_{y} \sum_{l=1}^{L} I(\hat{h}_l(x) = y)$$

where $\hat{h}_l(x)$ is the classification prediction of the response variable at x using the l-th Profit Decision Tree. CPRF predicts
the class at x that receives the most votes from the individual trees.

tive Profit Random Forests (CPRF), which mainly based on the combination of several independent binary trees. These random forests algorithms use the different way to build decision trees, which decides the differences between them.

Both RF and ROF employ the traditional node-split method to search the best split point by traversing the possible split point of each attribute variable, but the different is that ROF mainly applies transformation method (PCA) [27] to transfer the data at each node to another space when computing the best split at this tree node. According to the work by Raileanu et al., [36], there is no significant difference between the two node-split methods of Gini index and information gain ratio from the perspective of the performance of random forests algorithm, so we employ one of the node-split methods to split the tree node in RF and ROF, i.e., information gain ratio. In addition, ADF uses the node-split method of the weighted entropy to achieve alternate between overall weight updates and parallel randomized tree growing. Beyond that, unlike RF, ROF, ADF and CPRF, BRF first considered the evaluation of the power of attributes variables, which selects the most powerful attribute for each tree node, and finally the midpoint value of the most powerful attribute is chosen as the split point.

TABLE 1. Ten datasets used in the experiments.

Datasets	No.examples	No.attributes	No.classes
Yale32 × 32	165	1024	15
sonar	208	20	2
20Newsgroups	16242	100	4
isolet	6238	617	26
pima	768	8	2
housing	506	13	2
ionosphere	35	34	2
wine	178	13	3
waveform21	5000	21	3
musk2	6598	166	2
satimage	6435	36	6
breast-cancer	569	32	2
movement-libras	360	90	15
land-cover	1182	147	9

1) DATASETS

Fourteen real-world datasets are used in our experiments, which are Yale32 × 32 dataset, 20Newsgroups dataset and eleven datasets from the UCI Machine Learning Repository. Apart from image dataset (Yale32 × 32[1] and sonar), we also test our algorithm using data from text classification, speech recognition, medical diagnosis, housing values analysis, life and Physical sciences. Table 1 shows the characteristics of these datasets. The feature dimensions of these datasets vary from tens, hundreds, to thousands.

2) PARAMETERS SETTINGS

In our experiments, each random forests algorithm is composed of $L = 100$ trees, and the parameter L can be viewed as an indicator of the operating complexity of the algorithm. As the purpose of this study is to investigate whether our split node criterion contributes to improve the performance of the random forests, we decided to train all random forests with the same medium size. As long as the parameter L is the same for each algorithm, we can fairly compare their performances. At the same time, in order to build a decision tree, $m = round(log_2(p) + c) \ll p$ features are randomly selected, where p is the dimensionality of the data samples and $c \in R$ is a parameter. In addition, the ADF algorithm usually depends on certain parameters (e.g., the maximum depth D_{max} of the trees, the number of random thresholds of each node and the minimum number of samples for further splitting), and the choice of these parameters can have an impact on the performance of the algorithm, especially the maximum depth D_{max} of the trees.

For all the random forests algorithms, 5-fold cross validation is applied to select the parameters including the maximum depth D_{max} of the trees in ADF. We set D_{max} to either 10, 15, or 25, and choose the best one according to the results of 5-fold cross validation. In addition, according to the authors [30], other parameters of ADF are set to the default values, and the Tangent loss function [37] is employed in ADF. In addition, for the "isolet" dataset where the training and testing partition are given a priori, then we just use the training data to train the model and put all the testing data

[1] http://vision.ucsd.edu/ leekc/ExtYaleDatabase/Yale Face Database.htm

TABLE 2. Mean classification accuracy and standard deviations are obtained by the compared random forest algorithms. Algorithms with the best accuracy are shown in bold.

Datasets	RF	ROF	BRF	ADF	CPRF
Yale32 × 32	0.5156 ± 0.0183	0.5067 ± 0.0147	0.4986 ± 0.0182	**0.6969 ± 0.0934**	0.5218 ± 0.0400
sonar	0.6840 ± 0.0914	0.6972 ± 0.1351	0.7088 ± 0.1361	**0.8297 ± 0.0291**	0.7213 ± 0.1588
20Newsgroups	0.7729 ± 0.0579	0.7789 ± 0.0370	0.6973 ± 0.0144	**0.9863 ± 0.0452**	0.7632 ± 0.0470
isolet	0.9259 ± 0.0000	0.9326 ± 0.0000	0.9628 ± 0.0000	0.9493 ± 0.0000	**0.9637 ± 0.0000**
pima	0.7461 ± 0.0482	**0.7670 ± 0.0416**	0.7617 ± 0.0250	0.5871 ± 0.0250	0.7539 ± 0.0132
ionosphere	0.9315 ± 0.0384	**0.9435 ± 0.0379**	0.9315 ± 0.0530	0.8742 ± 0.0136	0.9287 ± 0.0506
housing	0.6418 ± 0.0645	0.8228 ± 0.0734	0.7964 ± 0.1230	**0.8756 ± 0.0498**	0.8439 ± 0.0893
wine	0.9658 ± 0.0251	0.9482 ± 0.0474	0.9706 ± 0.0509	0.9600 ± 0.0224	**0.9835 ± 0.0153**
waveform21	0.8406 ± 0.0129	**0.8496 ± 0.0103**	0.7494± 0.0106	0.7500 ± 0.1369	0.8210 ± 0.0090
musk2	0.8546 ± 0.1204	0.7009 ± 0.1989	0.8960 ± 0.0503	**0.9980 ± 0.0045**	0.9147 ± 0.0304
satimage	0.8970 ± 0.0110	0.8979 ± 0.0097	0.8735 ± 0.0206	0.7250 ± 0.1046	**0.9061 ± 0.0278**
breast-cancer	0.9687 ± 0.0230	0.9648 ± 0.0189	0.9432 ± 0.0240	0.9012 ± 0.1491	**0.9849 ± 0.0505**
movement-libras	0.6583 ± 0.1158	0.4589 ± 0.0769	0.4530 ± 0.0830	0.4893 ± 0.0203	**0.7096 ± 0.0402**
land-cover	0.9793 ± 0.0374	0.9742 ± 0.0439	0.9595 ± 0.0673	0.8505 ± 0.0404	**0.9812 ± 0.0109**

on the algorithm to output the classification accuracy result. The values of attribute variables were scaled to [0, 1] for each dataset.

3) RESULTS

All results are presented in Table 2, the best classification accuracy is shown in boldface. We can see that CPRF outperforms other existing random classification forests algorithms on most datasets. In particular, the performance of CPRF is better than that of the other existing random forests algorithms on the application hand movement recognition. The hand movement Database movement-libras contains 15 classes of 24 instances each, where each class references to a hand movement type in LIBRAS. These hand movements are extracted from the video. In the video pre-processing, a time normalization is carried out selecting 45 frames from each video, in according to an uniform distribution. In each frame, the centroid pixels of the segmented objects (the hand) are found, which compose the discrete version of the curve F with 45 points. All curves are normalized in the unitary space. Each curve F is mapped in a representation with 90 features, with representing the coordinates of movement.

Under the different instances and classes, the attributes of the same class of hand movement are likely to have a certain interdependency. This is because different hand movement have different characteristics. CPRF can explore the dependency relation of attributes by selecting the split point that makes the two descendants sample space have the highest gains according the *efficient* property of Banzhaf power index. That is, under the condition of known class labels, our algorithm selected the split point that makes the two descendants sample space have the best ability of class discrimination by computing the dependency relation of attributes. We finally obtain that the attribute variable of each non-leaf node in a single tree is ''willing'' to cooperate together to construct a decision tree, at the same time, these attributes are combined together to built tree can presents a much stronger prediction. However, the traditional node-split method only analyzes each attribute in turn to measure its predictive power in terms of the class output, which ignores the relationship between attributes. The reason that CPRF produced the best result is due to the use of a new node split method based on cooperative game theory, which can better 'consider' attributes relation for learning objective.

In addition, we must notice that the CPRF underperforms the ADF on some high-dimensional data, such as Yale 32 × 32, 20Newsgroups, waveform21 and musk2. However, the performance CPRF are better than ADF on some medium or low dimensional data. ADF was inspired by the Boosting algorithm, which can achieve the minimum global loss by updating the sample weight during training. According, ADF may be more efficient than CPRF on the high-dimensional data. CPRF mainly focus on learning a new node-spit method based on the cooperation game theory, which doesn't employ any means to processing sample. CPRF is devoted to exploring the dependency relation between the sample attributes.

B. DISCUSSION

In order to understand the difference between the traditional node-split method and the proposed node-split method, a meaningful experiment is carried out in this section. In particular, the breast-cancer dataset is used in this experiment. We use the information gain ratio and the proposed node-split method to build a decision tree for RF and CPRF, respectively, based on a small group of attribute variables of breast-cancer dataset. Note that the traditional node-split method pays less attention to the intrinsic structure of the attribute variables and fails to find attributes with good discriminate ability as a group but are weak as individuals. As shown in Fig. 1, a single tree in an RF tends to choose the most relevant (strong prediction power) attribute using the node-split method of information gain ratio. However, it often pays less attention to the intrinsic structure of the attribute variables. For the experimental results, we found that the attribute F, G and J together have a good prediction power, i.e., the attribute F, G and J are highly interdependent with each other. Although the attribute F, G and J are redundant, not all these attribute are really redundant. As can be seen in Fig. 1, the attribute group (F, G and J) can be easily destroyed by the node-split method of information gain ratio. We cannot guarantee that our method retains all useful attribute groups.

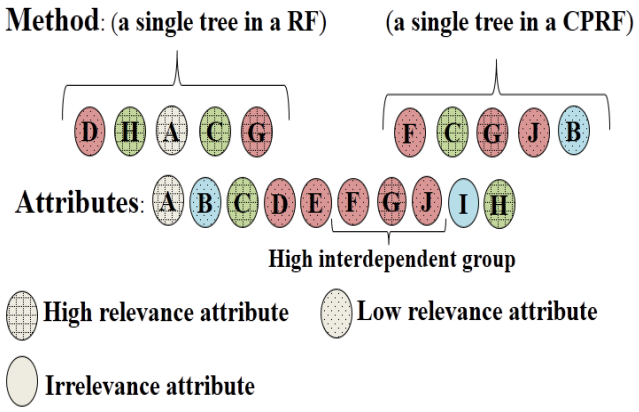


FIGURE 1. Single tree in an RF using the traditional node-split method, which disregards the interdependent group (Attributes of the same color means they are redundant to each other and the attribute *F*, *G*, and *J* are interdependent with each other).

However, the method can retain useful attribute group as many as possible. Note that attributes redundance and relevance are given by computing the mutual information and conditional mutual information [38].

C. ROBUSTNESS ANALYSIS

In order to investigate the robustness of CPRF to parameters, a robustness analysis experiment is performed. As shown in Figures, Fig 2 and Fig 3 show the performance of CPRF with different parameter m which controls the size of attribute subsets for each tree on two datasets (sonar and musk2) and with different parameter L which controls the number of decision trees in a random forests algorithm on several datasets, respectively. The robustness of the other datasets tell a similar story, so we here give some results on two datasets, as following.

In Fig. 2, we can see after m approaches 6 or 10 all methods lead to satisfying results for ''sonar'' dataset and ''musk2'' dataset, which demonstrates that CPRF is quite robust to the parameters (though the results tend to have small fluctuation) and thus it can save much time to tune.

In Fig. 3, we can see that CPRF is basically robust with the size L. With the increase of the number of trees, the accuracy increases gradually. In particular, when the number of trees approaches $L = 500 or 1000$, CPRF has a slightly higher accuracy rate. It indicates that CPRF will not incur over-fitting.

D. OCEAN FRONT RECOGNITION

With the gradual improvement in spatial resolution of remote sensing (RS), the high-resolution satellite imagery data are becoming increasingly available and inexpensive. From the high-resolution satellite imagery data, we can learn the remote sensing images with detailed information related to spatial arrangement information and textural structures. Ocean fronts have attracted a great deal of researcher interest. Most researchers only focus on the ocean front detection problem. The most popular methods include the gradient algorithms, the edge detector and entropy

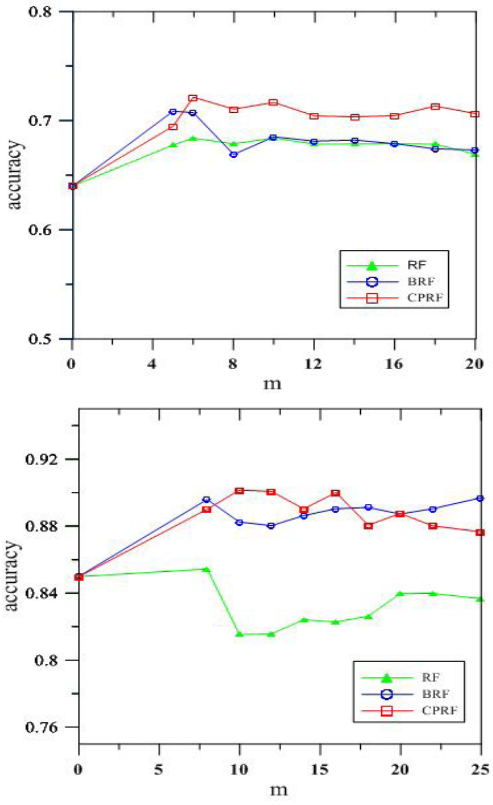


FIGURE 2. Performance of the random forests algorithms with different "*m*" parameter for "sonar" and "musk2" datasets. In the two charts, the y-axis shows the accuracy and the x-axis shows the different "*m*" values.

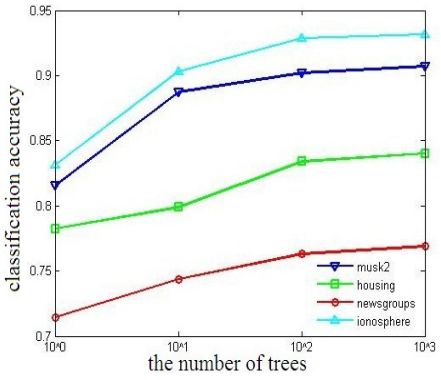


FIGURE 3. Performance of CPRF is influenced by the number of trees on four datasets, and in this plot, the y-axis shows the classification accuracy, and the number of trees is shown along the x-axis.

algorithms [39], [40]. However, these methods have lower efficiency for ocean front detection. Thus, we first try to recognition the front and non-front region from remote sensing images, then ocean fronts will be detected efficiently. We will use the different random forest algorithms to classify the regions with front and without front (non-front).

1) DATA

The sea surface temperature (SST) images data is used to study the ocean front recognition. The SST images obtained

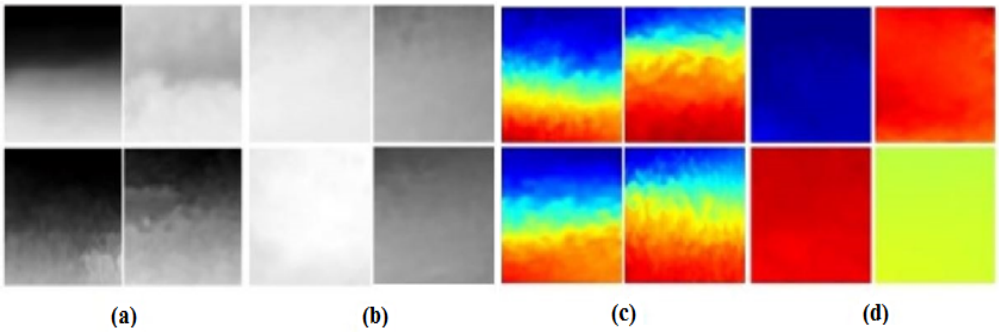


FIGURE 4. Examples of the front and non-front image patches. (a) The image patches of front. (b) The image patches of non-front. (c) Colorful image patches of front (a). (d) Colorful image patches of non-front (b).

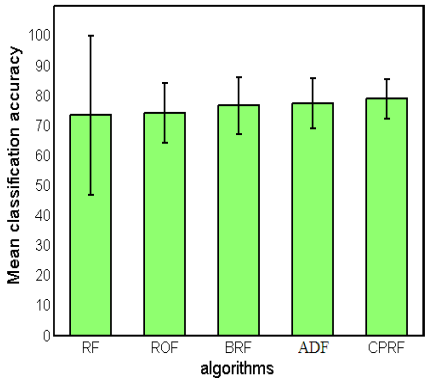


FIGURE 5. Classification accuracy of the different algorithms on the ocean front dataset.

from Advanced Very High Resolution Radiometer (AVHRR) sensor have allowed the oceanographers to study and monitor the mesoscale features in the oceans [39]. We first used the traditional edge detection method to detect the front by using (SST) images.[2] Based on the detection results, we crop some image patches from the SST images. These patches are labeled the front and non-front. To verify the accuracy of detection results, the patches are validated by a professional oceanographer.

In our experiment, we crop a total of 214 image patches from two subjects. Here, we give some image patch examples in Fig. 4, which were collected from National Oceanic and Atmospheric Administration (NOAA). Fig. 4(a) shows the front image patches. The non-front image patches are presented in Fig. 4(b). To more clearly exhibit the front and non-front, the color image pathes collected from the NOAA are shown in Fig. 4(c) and Fig. 4(d). An ocean front is generally defined as boundary between two distinct water masses, which is shown in Fig. 4(c), where the blue and red represents two distinct water masses. However, there is no such boundary in Fig. 4(d).

For each image patch, we employ the well-known feature extraction method in image processing, i.e., the Gabor wavelet algorithm [41], to generate the features. we use four spatial scales and six directions for the Gabor parameters. A single image patch after filtered will be generated 24 image patches, then the mean and variance are used as the feature of each image patch. Therefore, the resulting Gabor feature is 48 dimensional for each image patch. In this experiment, 5-fold cross validation is applied to select the parameters for all the random forests algorithms. The classification accuracies and standard deviations are shown in Fig. 5. We can see that CPRF could more accurately identify front and non-front than other random forests algorithms.

[2]http://www.class.ngdc.noaa.gov/saa/products

VI. CONCLUSION

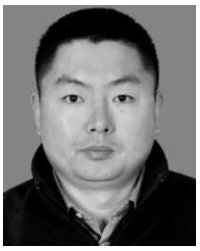
A new random forests algorithm called Cooperative Profit Random Forests (CPRF) is proposed. CPRF employs a new node-split method to develop cooperation between tree nodes of individual tree classifier in random forests based on the idea of cooperative game theory. The existing random forests algorithms are mainly based on the traditional split node method, i.e., the Gini index or the information gain ratio. However, the traditional method pays less attention to the intrinsic structure of the attribute variables and fails to find attributes with good discriminate ability as a group but are weak as individuals. Therefore, random forests employ the traditional methods, which will be difficult to obtain satisfactory results for some of the datasets with special intrinsic structure of the attribute variables. Experimental results have proven that CPRF can achieve the best performance on most real-world datasets, especially oceanographic data from remote sensing. In the future, we will combine deep learning model and CPRF algorithm to solve some practical problems in image processing and computer vision.

REFERENCES

[1] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proc. IEEE 11th Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.
[2] M. H. Pi, J. Ma, W. Zhang, and Z. Zhou, "Signal-walking-driven active contour model," *IET Image Process.*, vol. 9, no. 12, pp. 1101–1106, 2015.
[3] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Parallel Distrib. Syst.*, vol. 37, no. 8, pp. 1558–1570, Aug. 2014.
[4] M. Ristin, M. Guillaumin, J. Gall, and L. Van Gool, "Incremental learning of random forests for large-scale image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 490–503, Mar. 2016.

[5] K. Nordhausen, "Ensemble methods: Foundations and algorithms by Zhi–Hua Zhou," *Int. Statist. Rev.*, vol. 81, no. 3, p. 470, Dec. 2013.
[6] L. I. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees (CART)," *Encyclopedia Ecol.*, vol. 40, no. 3, pp. 582–588, 1984.
[7] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufman, 1993.
[8] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[9] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: A new classifier ensemble method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1619–1630, Oct. 2006.
[10] K. K. Kandaswamy *et al.*, "AFP-Pred: A random forest approach for predicting antifreeze proteins from sequence-derived properties," *J. Theor. Biol.*, vol. 270, no. 1, pp. 56–62, Feb. 2011.
[11] L. Zhang and P. N. Suganthan, "Random forests with ensemble of feature spaces," *Pattern Recognit.*, vol. 47, no. 10, pp. 3429–3437, Oct. 2014.
[12] J. N. Morgan and R. C. Messenger, "THAID: A sequential analysis program for the analysis of nominal scale dependent variables," Inst. Social Res., Tech. Rep. 3461, 1973.
[13] M. A. Muharram and G. D. Smith, "Evolutionary feature construction using information gain and gini index," in *Proc. EuroGP*, 2004, pp. 379–388.
[14] Y. Yan, J. Liu, X. Yao, and Y. Sun, "A nonrigid image registration approach based on cooperative game among pixels," *Syst. Eng. Electron.*, vol. 33, no. 5, pp. 1171–1175, 2011.
[15] I. B. Dimock and J. W. L. Wan, "Cellular image segmentation using n-agent cooperative game theory," *Proc. SPIE*, vol. 9784, pp. 97842T–97842T, Mar. 2016.
[16] H. Chen, S. Ta, and B. Sun, "Cooperative game approach to power allocation for target tracking in distributed mimo radar sensor networks," *IEEE Sensors J.*, vol. 15, no. 10, pp. 5423–5432, Oct. 2015.
[17] J. Jia, Y. Liu, and G. Zang, "A new algorithm for computing complex network reliability based on cooperative computing thought," in *Proc. 6th Int. Conf. Natural Comput. (ICNC)*, Aug. 2010, pp. 3120–3123.
[18] X. Sun, Y. Liu, J. Li, J. Zhu, X. Liu, and H. Chen, "Using cooperative game theory to optimize the feature selection problem," *Neurocomputing*, vol. 97, pp. 86–93, Nov. 2012.
[19] K. Zeng, K. She, and X. Niu, "Feature selection with neighborhood entropy-based cooperative game theory," *Comput. Intell. Neurosci.*, vol. 2014, Jan. 2014, Art. no. 11.
[20] A. Mortazavi and M. H. Moattar, "Robust feature selection from microarray data based on cooperative game theory and qualitative mutual information," *Adv. Inform.*, vol. 2016, 2016, Art. no. 1058305.
[21] D. A. Segar, *Introduction to ocean sciences*. Belmont, CA, USA: Wadsworth Pub., 1998.
[22] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.
[23] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832–844, Aug. 1998.
[24] Y. Amit and D. Geman, "Shape quantization and recognition with randomized trees," *Neural Comput.*, vol. 9, no. 7, pp. 1545–1588, 1997.
[25] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Mach. Learn.*, vol. 40, no. 2, pp. 139–157, 2000.
[26] S. W. Kwok and C. Carter, "Multiple decision trees," *Mach. Intell. Pattern Recognit.*, vol. 9. 1990, pp. 327–335.
[27] I. T. Jolliffe, *Principal Component Analysis*, vol. 87, no. 100. Berlin, Germany: Springer, 1986, pp. 41–64.
[28] X. Jiang, "Linear subspace learning-based dimensionality reduction," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 16–26, Mar. 2011.
[29] S. Bernard, S. Adam, and L. Heutte, "Dynamic random forests," *Pattern Recognit. Lett.*, vol. 33, no. 12, pp. 1580–1586, Sep. 2012.
[30] S. Schulter, P. Wohlhart, C. Leistner, A. Saffari, P. M. Roth, and H. Bischof, "Alternating decision forests," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 508–515.
[31] J. Sun, G. Zhong, J. Dong, and Y. Cai, "Banzhaf random forests," *CoRR*, vol. abs/1507.06105, 2015. [Online]. Available: http://arxiv.org/abs/1507.06105
[32] G. Chalkiadakis, E. Elkind, and M. Wooldridge, *Computational Aspects of Cooperative Game Theory* (Synthesis Lectures on Artificial Intelligence and Machine Learning). San Rafael, CA, USA: Morgan & Claypool, 2011.
[33] J. F. B. Iii, "Weighted voting doesn't work: A mathematical analysis," *Rutgers Law Rev.*, vol. 19, no. 2, pp. 317–343, 1965.
[34] P. D. Straffin, *Game Theory and Strategy*. Washington, DC, USA: Mathematical Association of America, 1993.
[35] D. L. Schilling, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2003.
[36] L. E. Raileanu and K. Stoffel, "Theoretical comparison between the gini index and information gain criteria," *Ann. Math. Artif. Intell.*, vol. 41, no. 1, pp. 77–93, May 2004.
[37] H. Masnadi-Shirazi, V. Mahadevan, and N. Vasconcelos, "On the design of robust classifiers for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 779–786.
[38] X. Sun, Y. Liu, D. Wei, M. Xu, H. Chen, and J. Han, "Selection of interdependent genes via dynamic relevance analysis for cancer diagnosis," *J. Biomed. Informat.*, vol. 46, no. 2, pp. 252–258, Apr. 2012.
[39] J.-F. Cayula and P. Cornillon, "Edge detection algorithm for SST images," *J. Atmos. Ocean. Technol.*, vol. 9, no. 1, pp. 67–80, 1992.
[40] J. K. Moore, M. R. Abbott, and J. G. Richman, "Variability in the location of the antarctic polar front (90°*irc*Ű20°*irc*W) from satellite sea surface temperature data," *J. Geophys. Res.*, vol. 102, no. C13, pp. 27825–27833, Dec. 1997.
[41] D. Gabor, "Theory of communication," *J. Inst. Electr. Eng. Jpn.*, vol. 93, no. 26, pp. 429–457, 1946.

JIANYUAN SUN received the B.S. degree in mathematics from Suihua University, Suihua, China, in 2006, and the M.S. degree from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 2013, where she is currently pursuing the Ph.D. degree in computer application technology with the Department of Computer Science and Technology. Her research interests include pattern recognition, image processing, and machine learning.



GUOQIANG ZHONG (M'16) received the B.S. degree in mathematics from Hebei Normal University, Shijiazhuang, China, in 2004, the M.S. degree in operations research and cybernetics from Beijing University of Technology, Beijing, China, in 2007, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China in 2011.

From 2011 to 2013, he was a Post-doctoral Fellow with the Synchromedia Laboratory for Multimedia Communication in Telepresence, Ecole de Technologie Superieure, University of Quebec, Montreal, Canada. Since 2014, he has been an Associate Professor with the Department of Computer Science and Technology, Ocean University of China, Qingdao, China. His research interests include pattern recognition, machine learning, and image processing.

Prof. Zhong is a member of the ACM, the CAA, the CAAI, and the CCF. He is also a member of the Technical Committee of the CAAI-PR. He received the Outstanding Reviewer Award from the *Pattern Recognition* journal for 2014 and 2015.

JUNYU DONG (M'09) received the B.Sc. and M.Sc. degrees from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, U.K., in 2003.

He joined Ocean University of China in 2004, and he is currently a Professor and the Head of the Department of Computer Science and Technology. His research interests include machine learning, big data, computer vision, and underwater vision.

HINA SAEEDA was born in 1989. She received the B.S. degree from UIITPMAS-AAUR Pakistan, in 2011, and the M.S. degree in software engineering from the Military College of Signals NUST, Pakistan. She is currently pursuing the Ph.D. degree in computer application technology with the Department of Computer Science and Technology, Ocean University of China, Qingdao, China. Her research interest is software engineering.

QIN ZHANG received the B.S. degree and the M.S. degree in computer science and technology from Shandong University, Jinan, China, in 2002 and 2005, respectively. She is currently pursuing the Ph.D. degree in computer application technology with the Department of Computer Science and Technology, Ocean University of China, Qingdao, China. Her research interests include pattern recognition, big data, and machine learning.