

A target-based color space for sea target detection

**Saeed Mirghasemi, Hadi Sadoghi Yazdi
& Mojtaba Lotfizad**

Applied Intelligence

The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies

ISSN 0924-669X
Volume 36
Number 4

Appl Intell (2012) 36:960-978
DOI 10.1007/s10489-011-0307-y

Volume 22, Number 2, March/April 2005
ISSN: 0924-669X CODEN APITE4

APPLIED INTELLIGENCE

*The International Journal of
Artificial Intelligence,
Neural Networks, and
Complex Problem-Solving Technologies*

Editor-in-Chief:

Moonis Ali

 Springer

Available
online
www.springerlink.com

 Springer

Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media, LLC. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your work, please use the accepted author's version for posting to your own website or your institution's repository. You may further deposit the accepted author's version on a funder's repository at a funder's request, provided it is not made publicly available until 12 months after publication.

A target-based color space for sea target detection

Saeed Mirghasemi · Hadi Sadoghi Yazdi ·
Mojtaba Lotfizad

Published online: 30 June 2011
© Springer Science+Business Media, LLC 2011

Abstract Sea target detection is a vital application for military and navigation purposes. A new supervised clustering method based on the combination of the PSO and FCM techniques is presented for the sea target detection problem. The color components of the target and non-target pixels in the RGB color space are used as features to train the classification algorithm. The new classifier is presented in the form of a new color space which we call the Target-based Color Space (TCS); in fact the RGB color space is converted to this new space through a 3×3 matrix. The Particle Swarm Optimization (PSO) algorithm is then used to search for the optimum weights of the conversion matrix which results in a more discriminating clustering space between the target and non-target pixels. In other words, solving the optimization problem, minimization of the objective function of the FCM clustering technique in linear and quadratic transform domain (with a NP-hard problem in quadratic conversion), is done using the PSO algorithm. The main objective of this work is to demonstrate the efficiency of using just color features, as well as color space conversion in the classification domain. Experimental results show the efficiency of new method in finding sea targets in color images.

Keywords Supervised clustering · Particle swarm optimization · Fuzzy C-means · Color space · Color

S. Mirghasemi
Islamic Azad University Parand Branch, Parand, Iran

H. Sadoghi Yazdi (✉)
Ferdowsi University of Mashhad, Mashhad, Iran
e-mail: h-sadoghi@um.ac.ir

M. Lotfizad
Tarbiat Modares University of Tehran, Tehran, Iran

segmentation · Linear and quadratic transformation · Sea target detection

1 Introduction

1.1 Sea target detection

Sea target detection methods through image processing, based on the type of images that have been used, are divided into four categories:

- (a) RADAR images
- (b) IR images
- (c) Satellite images
- (d) Visible images

Of which the first and most considered one concerns the sea target detection based on radar imagery. The radar system is essential for detecting in adverse weather conditions such as fog, raining, snowing, etc. [1]. Two significant kinds of radar images are SAR and ISAR images. Inverse synthetic aperture radar (ISAR) is an imaging technique which is used to produce medium/high resolution images (metric or sub-metric) of naval, aerial or ground targets. Most of researches in this area deal with radar imaging techniques rather than image processing techniques. One of the most considered parts of radar imagery are the SAR images. At present, there are two modes to detect ships in SAR images. One is a direct mode, detecting ships directly. The other is an indirect mode. That is, it first detects ship wakes, and then seeks ships around the wakes. In the direct mode, there are a lot of methods to detect ships, such as adaptive threshold way [2, 3], probability neural network (PNN) model method [4], double parameter constant false-alarm ratio (CFAR) detection method [4, 5], and fractal detection algorithm [6]. Ship wakes detection methods mainly include

Radon transform, Hough transform, mathematical morphology and wavelet analysis [7]. However, the wake of a slowly moving or nearly stationary ship is unlikely identifiable in a SAR image [8]. All the SAR-based methods expend largely but they have two serious drawbacks:

- (a) They can only obtain target points, which cannot be used to recognize the targets.
- (b) There are targets that are invisible to radar like wooden boats.

The second group consists of detection methods based on IR images. The RADAR images are often undesirable for military applications because they reveal the location of the imaging system. So, researchers explored visible and infrared images of ships which are generally more consistent than RADAR images and for which it is easier to compensate for environmental effects [9]. The infrared system is employed to enhance vision in weak light conditions. To extract semantic objects from a scene, an a priori knowledge of the image type and object characteristics is necessary. The a priori knowledge for the infrared image segmentation is based on the fact that the object that has to be extracted has either a larger or smaller temperature than the environment, being characterized by transient elements such as edges and peaks. The work published in [10] is one of recent years papers that worked on IR images. The authors used PCA, Bayes classification and wavelet-denoising to classify the sea targets, but in several papers, the limitations and disadvantages of methods based on statistical analysis are pointed out [11]. IR images have three main problems [9]:

- (a) Poor SNR-ratio
- (b) Varied gray levels
- (c) Naval ships or small boats have a homogenous stern side.

The third group consists of methods that use visible satellite images for sea target detection [12, 13]. One part of the last group is satellite high resolution images. High-resolution images allow a more accurate and reliable discrimination of ships and the possibility of their classification. However, the detection technique for this case must deal with a larger level of non-homogeneity and must take into account the intrinsic non-Gaussian nature of the back-scattered intensity [13]. Hu et al. [13] present a method based on cumulative projection curve (CPC) to estimate the number of ships of small size, which is only efficient on special images of stationary ships along coastline. One of a few research works that uses color feature, from the Lab color coordinate system, in this area is [12]. They have presented a definition on the degree of overlap between two clusters and developed an algorithm for calculating the overlap rate. Using this theory, they also have developed a new hierarchical cluster merging algorithm for image segmentation and apply it to the ship detection in high resolution image.

The last group consists of methods that work on visible images. Most of these methods use grey-level features for target detection [11, 14]. One of the research works that is more superior to previous works in this group is [11]. Their work is based on calculating different chaos by obtaining largest Lyapunov exponent of target and sea background which is not appropriate for images that contain some low chaos object other than the target. Also the same authors have proposed the work in [14] based on the natural measure feature. Although the method's results are considerable for some images but it still suffers from previously mentioned imperfection and needs analyzing several frames for exact results. To review the considered methods, we provide Table 1 to discursively compare some important aspects of sea target detection in different methods. The assessments in the table are based on the proposed methods on the previously mentioned works and may change over future works.

Because of the above mentioned reasons, the existing methods suffer from one or more drawbacks, so it is still important to find new methods of detecting the target from its background. This paper is one of a few works that utilizes the color feature in the sea target detection domain.

1.2 Color spaces

The use of color in image processing is motivated by two principal factors. First, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Second, humans can discern thousands of color shades and intensities, compared to about only two dozen shades of gray [15]. The purpose of a color space (also called color model or color system) is to facilitate the specification of colors in some standard generally accepted way [15]. Some papers like [16] present general segmentation methods which can operate based on information about motion, color, texture and so on, which indicates the importance of color as one of usable features for pattern recognition problem.

A color model is an abstract mathematical model describing the way colors can be represented as tuples of numbers, typically as three or four values or *color components* (e.g. RGB and CMYK are color models). Many color spaces are related to each other by linear transformations that are captured by 3×3 matrices. Hence a given color, and thereby any color image, can be represented in terms of another color space by transforming its 3-D vector representation using a 3×3 matrix. For instance, the calculations performed in the color space conversion from RGB to YCbCr are presented below. Parcels of each R, G and B input components are considered in the calculation of the output components in the space YCbCr [17]. The YCbCr color space was developed as part of ITU-R BT.601 during the development of

Table 1 Comparison between some important aspects of different sea target detection methods

	Detection of point-like targets	Military acceptance	Cost of making the image	A prior knowledge about the scene
SAR images	Yes	No	High	No
IR images	Yes	Good	High	Yes
Satellite image	No	Good	High	No
Visible images	No	Good	Low	No
TCS	No	Good	Low	Yes

world-wide digital component video standard. Y is the luminance component, and Cb and Cr are the blue-difference and red-difference chroma components.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

CIE standardized color order systems by specifying the light source, the observer and the methodology used to derive the values for describing color. Many color spaces are presented but the other viewpoint involves theories of color vision which are derived from the sums and differences of the three cone types [18–20]. Another viewpoint on color space is selection of the best space. In this category, different transformations of the RGB color space (as HSI, HSV and Lab) were compared to find the best method for separating target/clutter or foreground/background and so on in color images taken by a digital photo camera [21]. Some researchers used a color space in a specific application in the computer vision such as skin detection in the HSV color space [22], people tracking by using color histograms from the H and the V components of the HSV color space [23], face recognition in a new space based on K-L¹ transform [24], lip tracking into the CIELAB, CIELUV color spaces [25], object recognition into illumination invariant type of RGB color space [26] and image restoration into the CIELAB color space [27]. In [28], the properties of six color spaces are discussed for detection of specific surface defects. They show that these defects are well detected when the clustering analysis is performed in the RGB space. Authors of [29] apply two classification methods using different color spaces. By means of a visual assessment of the results, they also conclude that the RGB space is the best among all the considered color spaces.

The main advantage of segmentation through the color features is that target detection can be performed indepen-

dently of the size and position of the target within the image [30]. In most cases, the segmentation of a color image demonstrates to be more useful than the segmentation of a monochrome image, because the color image exhibits much more image features than the monochrome image [31]. The segmentation of a color image requires a computational cost which is considerably higher than what is needed for the monochrome image, but it is no longer a major problem with the increasing speed of computation and decreasing cost of color sensors. In fact, there has been a remarkable growth of techniques for the segmentation of color images in the past decade.

1.3 The classification concept

The field of machine learning falls into two primary categories: supervised and unsupervised learning. In supervised learning, the algorithm is provided with the both cases; data points and the labels that represent the concept to be learned for each case. The goal is then; learn the concept in the sense that a new, unseen case comes to be classified, the algorithm should predict a label for this case. The supervised segmentation is commonly used in the applications where the sample of object colors can be acquired in advance, e.g., object tracking, face/gesture recognition, and image retrieval etc. In order to avoid overfitting, these algorithms try to achieve a balance between fitting the training data and good generalization, this is usually referred as the Bias/Variance dilemma. The outcomes of this class of algorithms are usually evaluated on a disjoint set of examples from the training set, called the testing set. The common techniques of supervised segmentation are evaluated in [32]. The most common methods in this field are traditional statistical approaches, maximum likelihood, decision tree, nearest neighbor, support vector machine (SVM) and neural networks. Some examples among applications for supervised clustering are proposed in [33, 34] for target detection, and in [35] for building recognition.

The nature of the clustering problem is such that the ideal approach is equivalent to finding the global solution of a non-linear optimization problem. This is usually a difficult task to achieve. As a matter of fact, this is an NP-hard problem, i.e., a problem that cannot be solved in polynomial time [36].

¹Karhunen-Loève (KL).

The main goal of this paper is to create a new target based color space based on supervised classification, in which some characteristics of clusters to be classified are introduced to the classifier through human interference. In our supervised classifier, sample pixels from target and non-target are manually extracted (in form of a 25×25 sub-image for each one), and then the color components of these pixels are used to train the method. Thus, we make effort to gain a new conversion-based space with more discriminating characteristics between the color components of the target and non-target.

1.4 Related works

As mentioned, many research works have been published, utilizing color features. Here we are going to review some of them that have created new color space for target detection purposes. Several color spaces are developed for skin and face detection [37–39]. De Dios et al. [38] presented a new color space, YCgCr, using the smallest color difference (G-Y) instead of (B-Y) which is the biggest difference and is selected in YCbCr. A new color space and a level set method based on Mumford-Shah model for skin region segmentation is performed in [39]. Another new color space is constructed for skin color clustering in [37], and the concept of correlation between the image data and the color of illuminants is introduced.

Other applications also had been excuses for creating new color spaces. For example, a new color space for bladder tumor detection is proposed in [40]. In this new color space, they applied a non-linear transformation to the fluorescence component which leads to a better separation of fluorescing and non-fluorescing pixels. Jia et al. [41] presented a thresholding approach based on a new color space model and embedded it into the background subtraction. An approach to detect specular highlights in color images is presented in [42]. Panetta et al. [43] introduced a method of splitting up color spaces into different components and then performing edge detection on individual color planes. A new color space was also introduced which is an improved version of the PCA algorithm. By analyzing the results of these algorithms, they determined which color space and edge detector is best suited for each algorithm.

The FCM method is widely used in biomedical engineering for detection of infarct lesions [44], detection of breast cancer [45, 46], lungs nodule detection [47], analysis of lymph node sections [48], and other applications like ship detection in SAR images [49], road seed extraction [50], fire detection [51], face detection [52], lip segmentation [53], and proposing a polynomial-based neural network for pattern classification [54].

This paper is organized as follows; Sect. 2 is dedicated to describing the new method. Linear and quadratic conversions are introduced in Sects. 2.1 and 2.2 respectively. In

Sect. 2.3 the PSO procedure during our method is expressed. Section 3 is dedicated to experimental results where our new color space is tested from different points of views. A quick review of the proposed method and the innovations of this paper are mentioned in Sect. 4, and a brief introduction about the general PSO is mentioned in Appendix.

2 The proposed approach

In this section, we introduce a linear, as well as a quadratic conversion based on a novel combination of the PSO and FCM methods to create a target-based color space. We are seeking a 3×3 conversion matrix, and then using it to transfer the original image to a new color space. The conversion should have the quality of classifying target from not-target in the new color space. Therefore, our major criterion in finding the conversion matrix is differentiation between the color components of target and non-target pixels as much as possible, when the image is transformed to the new space. In other words, in the new space, we want the color component of target and non-target with maximum difference and minimum variance. To do so, we use the FCM as a rudimental method to cluster target and non-target, and then we try to improve this clustering performance. The PSO search is used as an iterative procedure to gradually optimize the conversion matrix to achieve better classification results. The details of our PSO based search toward finding the conversion matrix is discussed in Sect. 2.3. See also [55] for a similar conversion-based method through a 3×3 matrix, where we used quadratic programming and GA to produce a new color space for lip detection. Based on our previous experiments in [55] we realized that color feature could be a powerful tool to train supervised classification methods. Two methods are proposed in this paper to obtain the mentioned conversion matrix; linear and quadratic transformations. The proposed method should be trained in order to meet our conditions. Thus, at the first step, the color characteristics of the target and non-target pixels should be introduced to the software. In order to train our method, we extract sub-image pictures from the target and non-target pixels of the original image. These two sub-image pictures are called training samples and are used for training the method. Simply saying, the conversion matrix is obtained according to converting of the training samples, and it will be modified in the PSO procedure to make these training samples look more different after conversion. However, at the end of the PSO search the 3×3 matrix will be used to convert the whole image to the new color space.

2.1 Linear transformation

Different stages of our method are shown in Fig. 1. The first step is the extraction of training samples by choosing pieces

of target and non-target pixels from the original image. At the second step, the particles in the PSO are needed to be initialized in order to suggest a random conversion matrix, named W . So, the third step which is the linear transformation needs two inputs; the first input which consists the training samples and the second one that is the 3×3 conversion matrix. Then, we convert the samples to the new space by means of the matrix suggested by the PSO through the linear transformation represented as the following equation,

$$y = x \times W \tag{2}$$

where x is the training samples from the target and non-target that are arranged below each other in the form of a $M \times 3$ matrix. M is the summation of pixels in both training samples, and 3 denotes the number of color components in the original color space (RGB), where we have R, G, and B as color information. In the tested images throughout this paper, both the training samples from the target and non-target are in the form of 25×25 pixels sub-image. Consequently, their matrix would be a 3D, $25 \times 25 \times 3$ matrix in which 3 stands for R, G, and B. Therefore, the joined training sample (matrix x) is a 1250×3 matrix (625 pixels from target picture and 625 pixels from non-target picture). y is the corresponding converted information in the new color space, and W is the conversion matrix. In (2), x is a $M \times 3$ matrix in the original space. Hence, the number of rows in W must be 3. Then, the conversion matrix should be a $3 \times n$ matrix, where n denotes the number of color components in the new color space. Since, we wanted to stay loyal to the number of color components in the original color space, we retained n at 3 too. That is one of several reasons for setting W as a 3×3 matrix. Other reasons are mentioned in their appropriate place in the paper.

At the fourth step, we use the FCM method to cluster the achieved data in the new space. Our FCM clustering problem is based on the minimization of the following objective function:

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^m \|y_k - v_i\|^2, \tag{3}$$

$$\sum_{k=1}^n \sum_{i=1}^c u_{ik}^m \|(x \cdot W)_k - v_i\|^2, \tag{4}$$

where $Y = \{y_1, y_2, \dots, y_n\}$ is a finite subset of an S -dimensional vector space over the real data, Y denotes the set of feature vectors and c is the number of clusters, and $m > 1$ is the fuzziness index. The matrix, $U = [u_{ik}]_{c \times n}$ is called a constrained fuzzy c partition of Y if the entries of U satisfy the following,

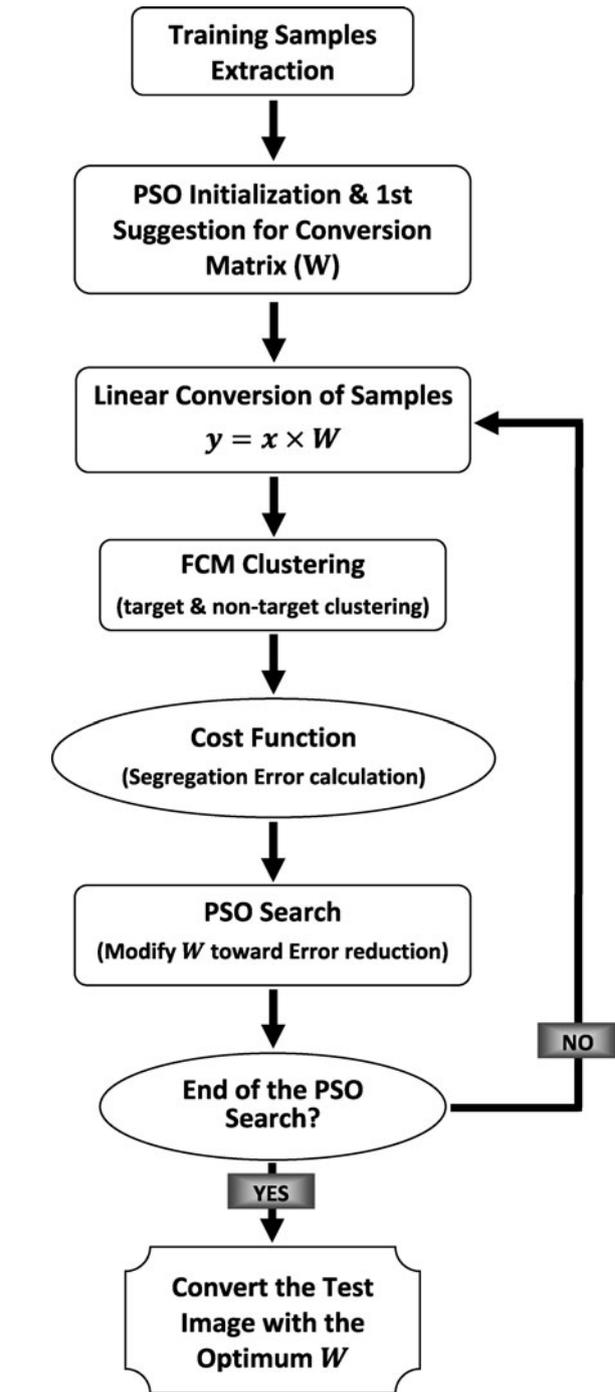


Fig. 1 Flowchart of creating the new color space

$$\begin{aligned}
 &u_{ik} \in [0, 1], \quad i = 1, 2, \dots, c \quad k = 1, 2, \dots, n \\
 &\sum_{i=1}^c u_{ik} = 1, \quad k = 1, 2, \dots, n \\
 &0 < \sum_{k=1}^n u_{ik} < n, \quad i = 1, 2, \dots, c
 \end{aligned} \tag{5}$$

and u_{ik} is the degree of membership of the k -th data item to the i -th cluster, $V = \{v_1, v_2, \dots, v_c\}$ is the cluster prototypes set and $v_i \in R^s$ is the center of the i -th cluster.

Next, in order to enhance the clustering execution, and to obtain the optimum transformation weights in the conversion matrix, we calculate the clustering error of the transformed data according to the FCM clustering parameters in a function called Cost function. Then, the PSO search method is utilized to reduce this error step by step in a determined number of iteration levels. In fact, the PSO searches a 9-dimensional space (because we have 9 weights to be determined) for the optimum weights, which would result in a W with more discriminating effects. More details about operation of our method are discussed in Sect. 2.3.

2.2 Quadratic transformation

Although the previously obtained W is powerful in the clustering domain, nevertheless in the following materials we will show that there is a better approach to find the weights of W , in a way that classification is more distinctively done, especially in images that targets and non-targets have close color contents. Here, the flowchart is the same as Fig. 1, except for the linear conversion which is replaced with the following quadratic conversion,

$$y = 0.5 \times (W \times (x \times W)^T)^T + x \times W \tag{6}$$

In a more simple form it becomes:

$$y = x(WW^T/2 + W) \tag{7}$$

Equation (6), including the constant coefficient, 0.5, is originated from the general form of quadratic programming topic [56]. This equation has a non-linear structure and consists of two parts. The first part has a $W \times W^T$ term which is a symmetric 3×3 matrix and has greater weights than the W alone. This makes the quadratic conversion more efficient in the transform operation. The nature of (7) is another reason for having a 3×3 conversion matrix. If we had used a $3 \times n$ conversion matrix in which $n \neq 3$, we would not have been able to use the quadratic conversion equation. That is because we cannot sum up the matrices having different dimensions. It is worthy to mention that there exist several two-part conversions with non-linear intrinsic that could lead into better results, but one of the most general ones could be (6).

Again the FCM method is used to cluster the obtained data in the new space. Therefore, we have to minimize the mentioned objective function in (3) with a different y that is obtained from (6). By rewriting the objective function, we

have

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c u_{ik}^m \|(0.5 * (W * (x * W)^T)^T + x * W)_k - v_i\|^2 \tag{8}$$

The common solution for solving this non-linear problem is through the Lagrange multipliers. To optimize the final solution, it is sufficient to solve the following set of equations,

$$\begin{cases} \frac{\partial J(U, V)}{\partial W} \\ \frac{\partial J(U, V)}{\partial x} \\ \frac{\partial J(U, V)}{\partial v} \end{cases} \tag{9}$$

It is clear enough that by expanding (7), due to the power of 2, we will have a non-linear equation with powers of 3 and 4. So the new objective function is non-convex, and it is hard to be solved through the Lagrange multipliers. In fact, the former equation due to its non-quadratic structure does not seem to have a routine solution. A common solution for these kind of problems is a comprehensive search that exposes the problem with so many answers, and tries to find the best and closest one by the trial and error method. One such algorithm is the PSO search method in which at first a random guess-work is made and then the procedure tries to converge it toward the real answer. Although, the final solution may not be the exact solution, but it is very close to it. Here, in our problem, we try to guess the W and gradually update it in a way that results to an optimal conversion matrix. Then, utilizing the PSO approach seems to be one of the best options to solve this non-linear non-convex problem. In the next subsection the details of the PSO search method in finding the optimum W is discussed.

2.3 PSO based procedure to find optimized conversion matrix

For a quick review of the PSO algorithm see the Appendix at the end of this paper. The following pseudo code describes our method in finding the best clustering conversion matrix:

Comment: *PSO-based optimizing conversion matrix problem algorithm,*

Load each R, G and B color values from Target training Sample separately.

Load each R, G and B color values from non-Target training Sample separately.

Put these values along each other as a single matrix;

Comment: *first, values from target and then values from non-target.*

Comment: *Define the solution space, fitness function, and population size,*

For each W ,

Initialize:

- a: Minimum and maximum value of particles velocity in each dimension $[v_{min}v_{max}]$; **Comment:** normally a number between -0.5 and 0.5 .
- b: Maximum iteration, number of particles; **Comment:** usually 100 and 50 respectively.
- c: Number of dimensions; **Comment:** for testing sea target images it is always 9.
- d: Minimum and maximum value of particles value in each dimension $[x_{min}x_{max}]$; **Comment:** usually a value between -1 and 1 .
- e: Initial and final value of Inertia weighting factor, w ; **Comment:** here, 1.
- f: Determine velocity values matrices; **Comment:** according to Maximum iteration and Number of dimension.
- g: Determine initial value for particles x ; **Comment:** according to number of particles and $[x_{min}x_{max}]$.

While Iteration <Max iteration or population uniformity < threshold

Update inertia weighting factor w ; **Comment:** according to (11).

Call Cost function; **Comment:** this function is constituted from two nested functions called Conversion and Error functions. Calculating the Error value is done here (Error value is the number of misclassified converted sample data after the FCM clustering).

Conversion function

Form W according to current w vector values
Perform linear or quadratic conversion;
Comment: according to (2) and (6) respectively.

Error function

FCM clustering
Error value calculation

End Error function

End Conversion function

Set restricting decisions to prevent x and velocity values from growing limitless.

Finding $pbest$ and $gbest$ for particles and their related x values;

Comment: $pbest$ and $gbest$ are best position for each and all particles respectively.

Update \vec{v}_i^{k+1} and \vec{x}_i^{k+1} ; **Comment:** According to (12), (13).

End While

End For

Display founded best \vec{x} .

Comment: final conversion on the whole original image.

Form final W according to obtained values for x particle with the $gbest$ position

Load the original image,

Separate each R, G and B values,

Convert these values with the same applied conversion in the Cost Function,
Form the new image with three converted values and display it.

This structure includes four main parts:

- (a) Extraction of training samples
- (b) The PSO algorithm
- (c) Cost Function
- (d) Conversion of the whole original image

2.3.1 Adapting the PSO algorithm for solving the problem

The PSO algorithm was described in the previous section and now we explain matching it to our problem. $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})$ is the i -th particle with n dimensions. In our problem, the conversion matrix, W , has 9 elements. Then here, the search space is 9-Dimensional ($n = 9$), so $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{i9})$, i.e., each particle represents one variable with 9 dimensions. Each particle is initialized with a primary location and a primary velocity. The number of particles as well as the number of iteration levels should be reasonable with respect to the usage. Typical numbers are mentioned in the experimental results. The general approach is that we fly the 9D particles in order to lead their weights in a sense that cover our criteria for the new space. The tuning scheme for this goal is the Cost function. The Cost function decides for the value of each particle, and the search is done based on the cost of all particles as explained in Sect. 2.3. During the PSO search these particles are updating their location and velocity in such a way that the final obtained matrix leads us to the solution of (4) or (8), which in other words represents the best possible separating conversion matrix according to the used training samples. The next sub-section includes the PSO Cost function description in finding the optimum W .

2.3.2 Cost function in the PSO algorithm

We need a function to tune up the velocity and position of our particles in the PSO search. This function is a criterion that connects the FCM and PSO in a way that the final result would be an optimum matrix, and eventually through the conversion would visually separate the target and non-target pixels. The Cost function is composed of two nested functions named Conversion and Error functions. Three main operations are performed in the Cost function:

- The conversion operation of the training samples.
- The FCM clustering of the converted samples.
- Calculating the Error value in the new space version of training samples.

After initialization of the PSO search and extraction of the training samples, they have to be converted to the new space. More clearly, the training samples are converted by the proposed W through the linear/quadratic conversion. This task is done in the Conversion function in which the W is formed according to current position of particles (x vector). Then the Error function is called inside the same Conversion function. In the Error function, the FCM clustering is performed and the number of misclassified data in the new space is calculated. There are three color components for the converted sample, so we have three Error values, the maximum of these values is fed back to the PSO search body, as the single output of the Cost function. In this way, we are trying to minimize the worst error of the conversion in each triple color component. That is, by reducing the largest value, we have definitely reduced other values as well. Another alternative is to sum up all three values and then return the total as the single output of the Cost function, but the final result is a little different, and anyway the max-of-three-values method is experimentally better. Thereby, increasing the number of color components in the new space would result in higher dimensional matrices, which is equivalent to more coefficients to be determined, and also additional dimensions to be clustered with the FCM, and this obviously is not what we are looking for because in this case the process would take more time to perform.

2.3.3 The error value calculation

In our algorithm the Error value is obtained by the Error function in which, by means of the FCM method we try to produce an error based on belonging of the transformed training pixels to the target or non-target clusters. In other words, we intend to calculate the clustering error of the performed conversion. Since the united training sample is formed from the joint of target and non-targets color values, they are not totally mixed. That is, the first half of the united matrix contains values from target and the second one is composed of value from non-target. We expect after the conversion, target values are still in the first half of the clustered matrix and non-target values are in the second half of it. Therefore, we divide the transformed pixels into two groups and try to count the number of pixels that are not in their own clusters. To do so, we use the degree of membership (u_{ik}), obtained from the FCM procedure, and see whether after the conversion, the target and non-target pixels are still separate (i.e. in two clusters) or not. Then, using the membership values, it is quite simple to determine if they are mixed after conversion or not. It is expectable that the lower the Error value is, the better the clustering is performed. So during the PSO execution the aim is to minimize the Error value which is described as follow,

$$\text{Error value} = e_{n1} + e_{n2} \quad (10)$$

where e_{n1} denotes the number of pixels that do not belong to the first cluster (target cluster) and e_{n2} denotes the number of pixels that do not belong to the second cluster (non-target cluster).

2.3.4 Velocity weight update control

This operation is carried out with a factor called the inertia weight. Many references have pointed out that good values for inertia weight are usually slightly less than 1. From our experiments in the PSO particle velocity control, we found out that choosing the inertia weight between 0.5 and 1 results in maximum velocity coincidence. Then, in the standard PSO, the inertia weight is introduced as a descending parameter which is initially set to a higher value, $w_{initialize} = 1$, and finally it becomes $w_{final} = 0.5$. A linear relation is defined per iteration according to:

$$e = (w_{initialize} - w_{final}) \frac{(k_{max} - k)}{k_{max}} + w_{final} \quad (11)$$

where $w_{initialize}$, initializes the value of the weight factor and the final weight is w_{final} , k is the iteration number and k_{max} is the maximum number of iterations. In the first iteration, k is set to zero and so w is $w_{initialize}$, gradually w decreases to w_{final} . Choosing values out of this interval leads to an uncontrollable particle velocity which affects the best global solution.

3 Experimental results and discussion

For some reasons, the sea target databases, unlike face and skin images, are not usually in common access. The lack of famous visible color images for sea target detection, forced us to create our own database. We have successfully tested our new color space on 2 different databases. Our First database comprises about 2000 frames of the sea targets from several mainstream movies and about 100 pictures which are collected from the Internet. The second one is the database from low quality videos filmed by us for sea target detection purposes. This database is composed of images in which the contrast, resolution, brightness and chroma difference are generally low. We have chosen the poorer quality intentionally because this is the case for real sea target detection problems. Before reviewing the experimental results, it is worth mentioning that in the real sea target detection applications, the background of the image is mostly composed of sea and sometimes sky, and nothing more. So, expecting sea targets with complicated backgrounds would seem somewhat unreasonable. Hence, most of the used images in this work are sea targets with simple background, but for demonstrating the performance of our method even for complex pictures, a few images with complicated backgrounds

Table 2 Comparing the effect of the linear and the quadratic transform on some standard databases

Data Base	Number of Data Points	Conversion Matrix Size	Error Value in the First Iteration		Error Value in the Last Iteration	
			Lin. Conv.	Quad. Conv.	Lin. Conv.	Quad. Conv.
Iris (versicolor, virginica)	100	4×4	11	13	3	2
Wine (2,3)	119	13×13	38	38	38	37
Glass (1,2)	113	9×9	63	60	58	57

are also included. In this section, we will demonstrate the applicability of our method in different conditions and different applications. First of all, we test our method as a general classifier in non-image datasets. Secondly, the efficiency of the TCS is considered in the mentioned sea target databases from different perspectives. Next, the sample selection strategy, and the performance test of the proposed method are discussed. Finally, another application of the TCS in image classification is introduced.

3.1 Testing the proposed methods on non-image datasets

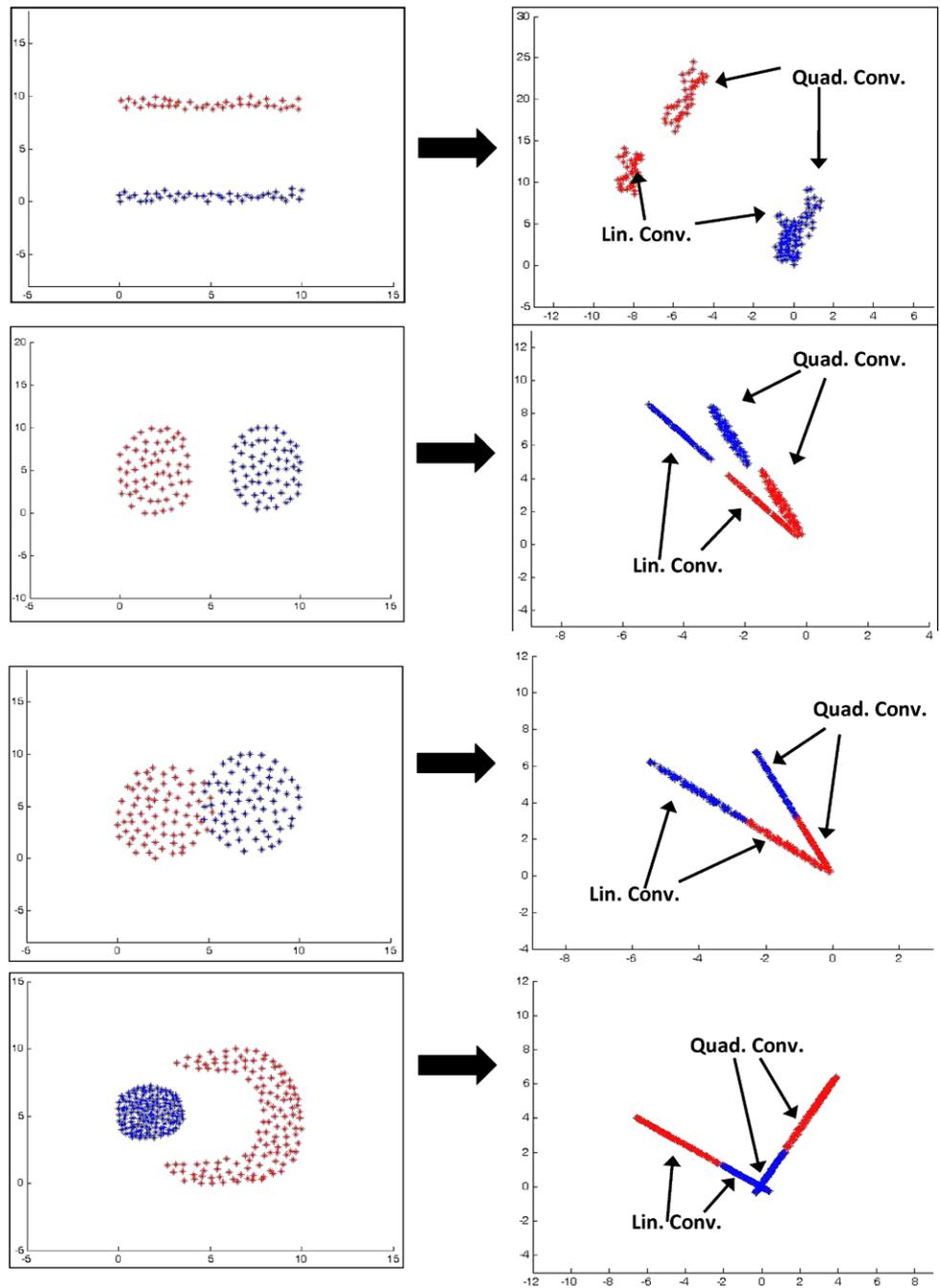
Discrimination performance of the proposed method is investigated using some standard datasets. These datasets are created as benchmarks to evaluate the validity of different classifiers. For example, the Iris dataset is composed of three different types of data, which are Setosa, Versicolor and Virginica. Each type has 50 data points, and the characteristic of each data point is specified with 4 numbers, which are the indicators that are used to train our method. Therefore, we need a 4×4 matrix to convert this dataset into a new space. The procedure for classification is as follows; first we convert the dataset by a random matrix, then we try to optimize the matrix in order to clearly discriminate between two clusters and reduce the mentioned Error value. More clearly, our effort in the new space is to evaluate the ability of our method in differentiating two categories of a dataset, while they are mixed in their original form.

Table 2 shows the results of testing our classifier with these datasets. Error value, in Table 2, indicates the number of misclassified data after converting to the new space. For instance, the final Error value for the Wine dataset in the quadratic conversion is 37. That is, the final quadratic conversion with the optimum W still has got 37 misclassified data in the new space. As depicted in Table 2, the quadratic transform always has lower classification error than the linear one. Moreover, our method is more efficient in low dimensional databases like Iris. According to these results when the matrix dimensions are increased, the number of misclassified data are increased as well. Therefore, the lower the matrix dimensions are, the lower the Error value we would have in the new space. This is yet another reason

why we kept the column number of W at the minimum, 3. The particles and iteration levels numbers are 50 and 100 respectively in the entire performed tests of Table 2. As explained, the conversion matrix for Wine and Glass datasets should be 13×13 and 9×9 respectively.

The second kind of experiment to reveal some aspect of our method, tests the proposed method on an artificial two-class dataset. This dataset is composed of two different classes which have been produced manually with different configurations in the Cartesian coordinates. The aim is to convert the points of each class through the linear and quadratic procedures and see if they are more distinctive in the new space. The Cartesian coordinates of each point are used as features to train our method. Since we have one value for the x -axis and one value for the y -axis, here our conversion matrix is a 2×2 matrix. After the conversion, we expect the points of each class, and correspondingly the two classes, to be moved in such a way that they seem more separate in the Cartesian coordinates. Different combinations of class1 (blue points) and class2 (red points), with the new space resulted from linear and quadratic transformations are depicted in Fig. 2. In each configuration, less than half of the total number of data points from each class is randomly chosen to train the method. Finally, the resulted conversion matrix is generalized to the whole data (used to convert the whole dataset). As it can be seen from Fig. 2, the method tends to compact the classes especially in one direction. Sometimes, like Fig. 2(b), the quadratic conversion has considerably better result, in which the transformed data has less entropy. This means, although the conversion may seem not particularly successful in rearranging the position of the two classes in a distant manner, but it is effective in reducing the entropy of each class, which is an efficient quality for the classes to look more distinctive. Another important point to mention is that, in Figs. 2(c) and (d), the two classes are somewhat mixed (they have common coordinate values for some points), but the proposed method tries to differentiate between the two classes and arrange them in a line. As before, the PSO tries to find the optimum weights for the 2×2 conversion matrix. The difference with the general case that we are looking for a 3×3 matrix is in the initialization of the PSO. Other than that, the procedure is the same.

Fig. 2 Comparing class2 data conversion in different configurations with the proposed linear and quadratic methods

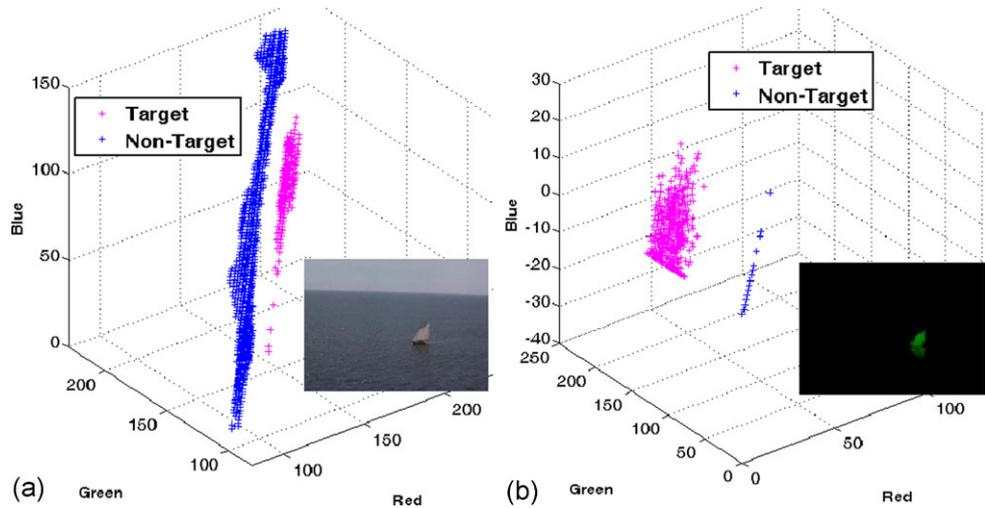


3.2 Testing the TCS on sea target image database

Here, the visual motivations for creating the new Target-based Color Space (TCS) are investigated. Figure 3(a) shows a 3D plot of target, and non-target pixels, in which each pixel is mapped into a 3D space according to its color component values. That is to say, according to values of triple color components of each pixel, a position in a 3-dimensional space is attributed to it. A 256×256 pixels image from non-target part of the image and a 25×25

pixels image from the target part of that image are used to show the distribution of color components in its original form (in the RGB color space) and then the same is done for the TCS. It can be seen from the original picture that the color components of these two classes (target and non-target) are somewhat close (especially the target and the sky from the background), and the position of pixels are varying over a long range. Figure 3(b) shows the converted image in the new color space. As it presents, not only the distance between the target and non-target pixels is further

Fig. 3 The plot of target and non-target pixel's color components in; (a) the original image. (b) The converted image



but also the entropy of each class is less. It should be mentioned that the second plot is zoomed for observing more details.

In the next step, we will show that none of the classical color spaces can meet the visual criteria for distinctive target classification. This is yet another motivation to create the TCS. As depicted in Fig. 4, the image in Fig. 4(a) is shown in four different color spaces. Figures 4(e) and 4(f) show the linear, and the quadratic conversions respectively. It is apparent from this figure that the quadratic conversion yields the best result from the distinctive segmentation point of view.

Our major problem in target detection lies in images in which the target and non-target have the same or close color contents. Existence of color differences between the target and non-target pixels is the basic assumption in our method. So, in images with exactly the same color components for the target and non-target pixels, the new color space is impotent, and the classification task is not possible. But, in images with close color contents which we call them Close Color Content (C^3) from now on, we have a solution which is the quadratic conversion. In contrast with linear transformation, the quadratic one has two considerable superiorities:

- (a) It results in less Error value after fewer iteration steps. It means, not only more minimum Error value is achievable through the quadratic conversion, but also a lesser number of iterations is required.
- (b) It is more efficient for segmentation in C^3 and other images.

Figure 5 compares the results of the Linear and Quadratic conversions for images from both mentioned databases. Because of the error reduction specification of the quadratic conversion, the final results for the Quadratic conversion are more distinctive. For Fig. 5(a), the linear transmission has

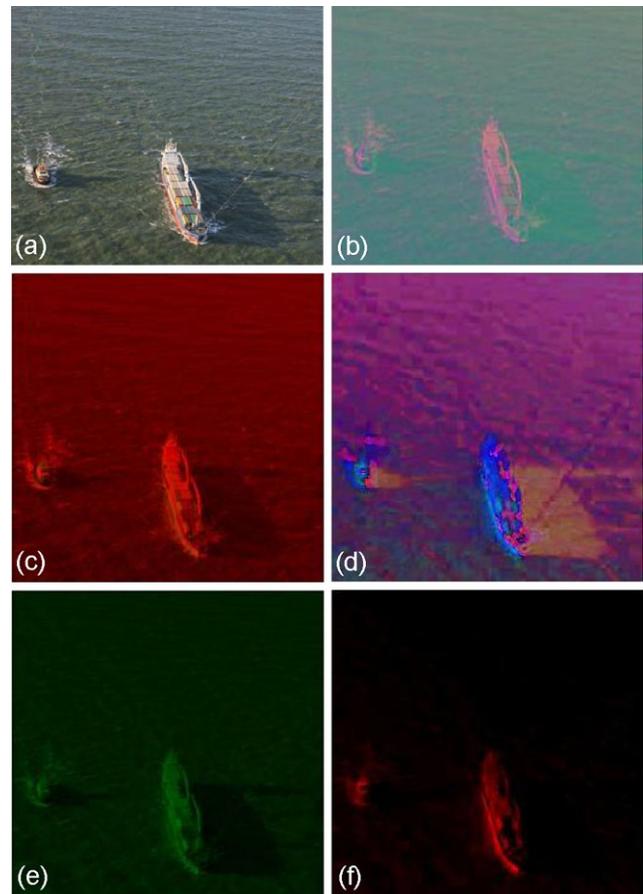


Fig. 4 The new color space in contrast with some classical color spaces. (a) Original image. (b) YCbCr color space. (c) YIQ color space. (d) HSV color space. (e) Linear conversion. (f) Quadratic conversion

the error value of 225, which means 225 pixels from 1250 pixels of training samples are misclassified. However, for the quadratic one, this value is 0. Besides, the quadratic transform converges to an Error value of 0 after 7 iterations, while

Fig. 5 Comparing the results of the linear and quadratic conversions. (a) The original image from the second database. (d) The original image from the first database. (b, e) The linear conversion results. (c, f) The quadratic conversion results

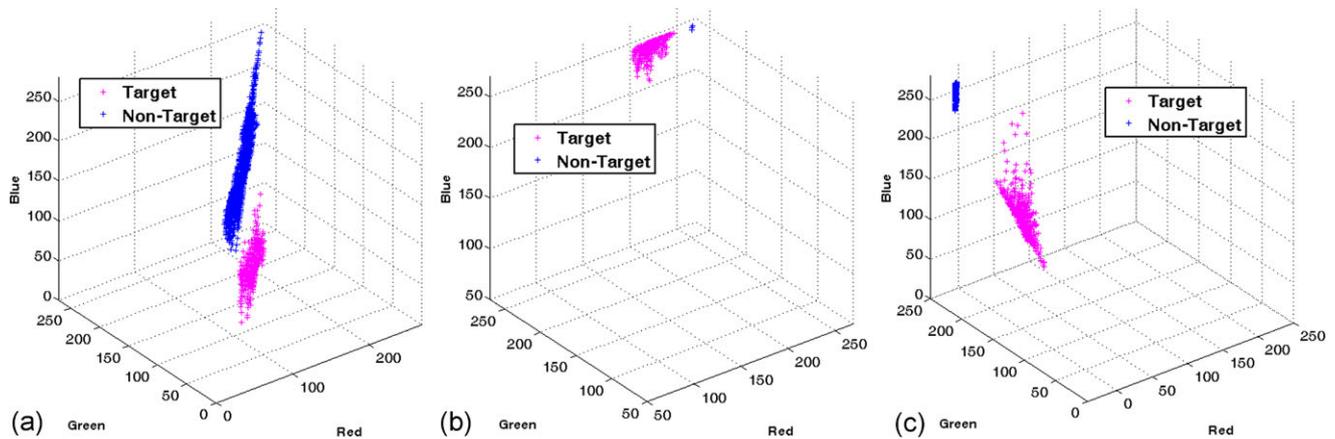
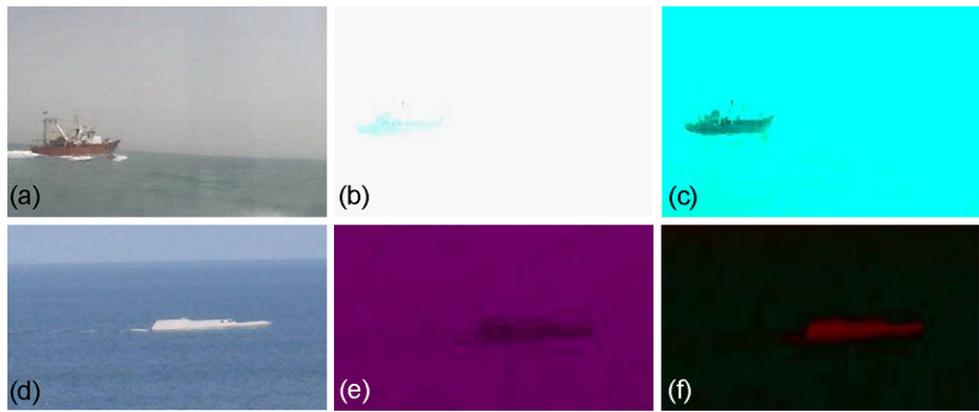


Fig. 6 3D plot for pixel's color component of images, (a) Fig. 5a, (b) Fig. 5b, and (c) Fig. 5c

the linear one could not reach a better Error value of 225 pixels, with the maximum iteration number of 100, and 50 particles. Figure 6 shows the 3D plot of the color components of the pixels in Figs. 5(a)–(c). The obtained conversion matrices for Figs. 5(b) and 5(c) are W_L and W_Q respectively:

$$W_L = \begin{bmatrix} 0.1768 & 0.4524 & 1.1236 \\ 0.7211 & 1.5524 & 2.1966 \\ 1.5137 & 2.8748 & 3.60452 \end{bmatrix},$$

$$W_Q = \begin{bmatrix} -0.1131 & -2.1594 & -1.2894 \\ 0.2451 & 3.1659 & 2.7297 \\ 2.2744 & 1.9851 & 0.9326 \end{bmatrix}$$

As depicted in Fig. 6, although the linear 3D plot entropy is better but because the quadratic 3D plot creates further distance between the target and non-target pixels, the converted image by quadratic method has yielded a better visual result. However, the quadratic conversion has a problem in its original form; it increases the intensity of color components for each pixel due to the nature of its structure. This feature biases the whole image toward bright colors like

white, light blue, light yellow, and etc. To solve this problem, we multiply color contents of each pixel with a factor having a value less than one. This factor for the obtained image in Fig. 5(c) is 0.5.

To compare and contrast another aspect of the TCS and the RGB color space, we have applied the Sobel edge detector to both images in Figs. 5(a) and (c), with results depicted in Fig. 7. This can be assumed as a post-processing technique, especially in cases where the final image is supposed to be processed by a machine, and not by a human-based visual system. Although the Sobel filter has the same threshold value for both images, but the resulted binary image from the RGB color space contains so many unwanted edges from the non-target, while all detected edges in the TCS color space are from the target.

To show the practicality of the proposed method, we segmented numerous images from the second database with one W , which is obtained through the quadratic conversion from the most left image in the first row in Fig. 8. Surprisingly, even for images which we do not know anything about, and no training samples are extracted from, the target

is still detectable. The obtained matrix is:

$$w = \begin{bmatrix} -2.7386 & 1.6952 & -3.4514 \\ 1.9795 & -1.9741 & 2.5340 \\ 1.3905 & 0.0321 & 1.8645 \end{bmatrix}$$

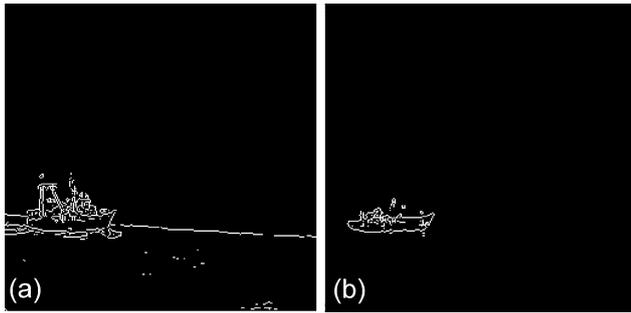


Fig. 7 Comparison of edge detection results between the RGB color space and the TCS. (a) Edge detection in Fig. 5(a). (b) Edge detection in Fig. 5(c)

For this experiment the PSO is initialized with 50 particles and 100 iteration levels. As depicted in Fig. 8, the above matrix has the same classification results for other images. Although the lighting, target and background conditions are somewhat different, using the same conversion matrix, the target is similarly segmented for all images. This feature is of interest especially in real-time applications. Of course, there are some pictures for which the classification results are not good enough, which are shown in the last row. In these frames, the lighting condition which is either too poor or too dazzling, prevents the sea target from looking visually different.

To reveal the efficiency of the TCS by means of the Quadratic conversion, and also taking preferences of using color features, some pictures with complicated background are selected from the first database, and the converted re-



Fig. 8 Obtaining a single conversion matrix for classification in the second database



Fig. 9 The efficiency of the TCS in some complicated pictures from the first database

sults are depicted in Fig. 9. The previous PSO initialization is again applied here.

To compare the TCS with other state-of-the-art algorithms, we compared the results of our method with one of the most famous classifiers, the Support Vector Machine (SVM), as follows. The results in Fig. 10 show that, although the SVM is more successful in preserving the exact shape of the detected target, but it has also detected some other parts of the non-targets as target.

3.3 Strategy for sample selection

First of all, it should be noted that different sampling from target figure cannot result in different sample pictures from the target. This is due to the fact that the target is a small part of the picture (usually less than 5% of the whole image), and selecting a 25×25 pixels image from it would cover almost the whole target picture. In other words, when repeating the sampling procedure, we cannot obtain another target sample picture with very different color components. But in non-target sampling the situation is different. To include the color diversity in the background or non-target part of the image, the non-target sample is selected from three different parts of the non-target picture. Again, if the background is simple, i.e. nothing more than sea and sky, different sampling would not change the color components of the non-target sample very much. Even when the non-target sample is slightly different, the results are visually the same, as shown in Fig. 11. In this figure, two different sampling procedures are performed, and then the proposed color space is tested on them. Here, the images of the related training samples are shown below the classification results. As a difference between two results, in the second sampling (Fig. 11(c)), the target in the classified image is a little more chromatic which is not tangible for human vision at all.

However, if we are dealing with complicated images which have a vast diversity of colors, the sampled non-target image in different sampling procedures would not be the same. Here, sometimes it is necessary to include four parts of the non-target part of the image, as shown in Fig. 12. Although, with a different sampling the classification results are different but, it has been done in an admissible manner. Consider the following images with the extracted samples, despite the fact that, the non-target samples are different but, the classification results seem acceptable in both cases.

Regarding the size of the applied training samples in the paper, as mentioned before, to establish a tradeoff between the time consumption and acceptable classification results, a 25×25 pixels size is chosen for training samples. In addition, we wanted to detect the target with the least dependency to the original image. Therefore, increasing the size of the training sample would increase not only the consumption time, but also the dependency of the introduced method to, to be classified image. However, an increase in the size of training samples would improve the classification performance.

3.4 Performance test

This section is devoted to evaluate the performance test of our approach. As mentioned before, having the conversion matrix obtained, the detection procedure is applicable to real-time applications. That is, the only work that should be done is simply multiplying the image and the matrix which is doable in a fraction of a second. Practicality of this idea is demonstrated in Fig. 8. But if the duration of the whole procedure, is of interest, it should be mentioned that in the whole procedure including running the PSO search and the FCM clustering, and continuing it till the final conversion matrix is obtained, the consumed time depends on three factors; the size of the training samples, the number of particles,

Fig. 10 Comparison of the detection results in SVM and TCS classifiers. (a, b) The original images. (c, d) The SVM results. (e, f) The TCS results

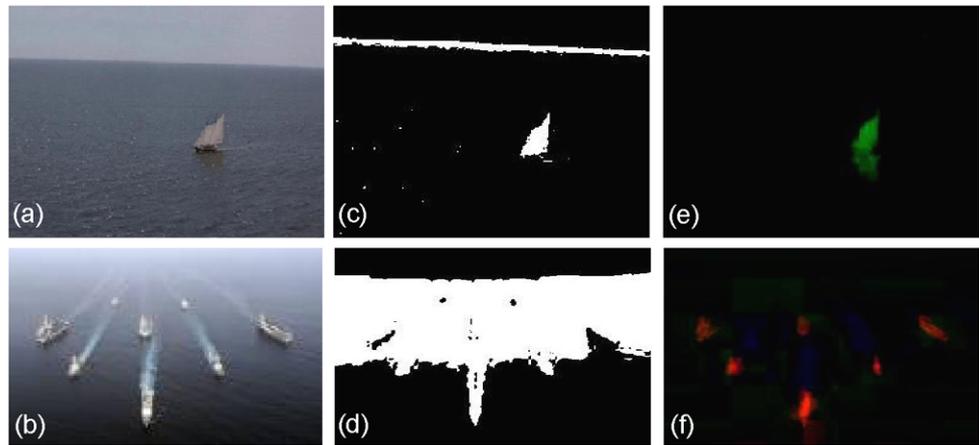


Fig. 11 A sample comparison of detection results while the training sample from the non-target in a simple background image is different. The used training samples are shown below each classification result. (a) The original image. (b, c) Detection results with the used training samples

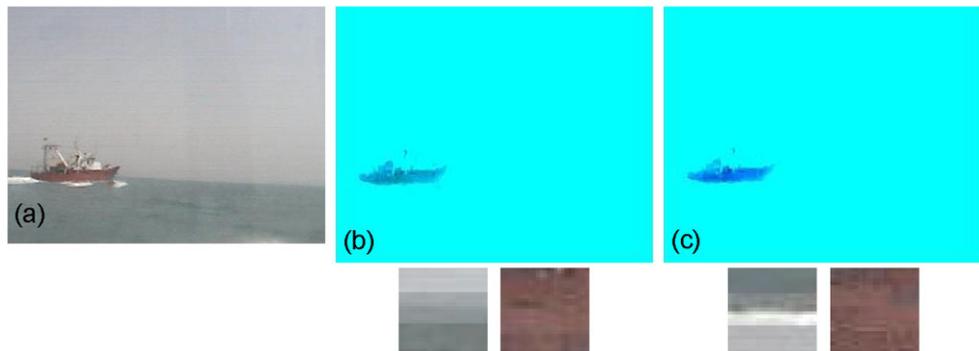
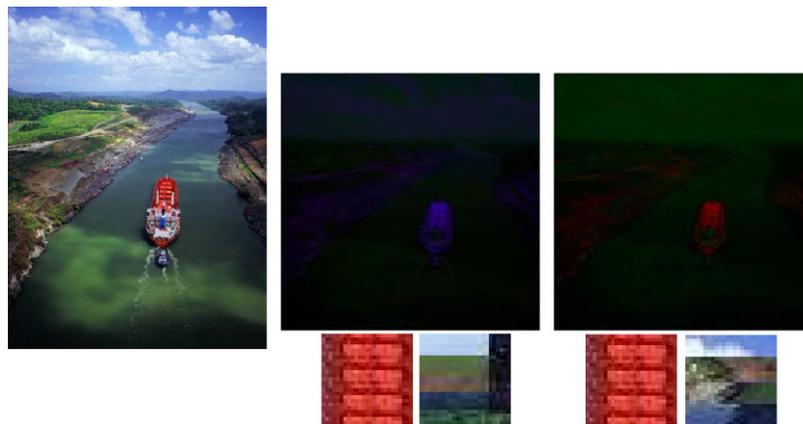


Fig. 12 A sample comparison of detection results while the training sample from the non-target in a complex background image is different. The used training samples are shown below each classification result. (a) The original image. (b, c) Detection results with the used training samples



and the maximum number of iterations. The larger these three parameters are, the more time is needed to attain the conversion matrix which is the cost for having more exact results. In Table 3, several tradeoffs between these factors, and the consumed time for each setting, in which the results are admissible, is shown. The type of hardware configuration will vary these measured times. The PC we used has an Intel core 2 quad CPU (Q6600@2.41 GHz) and 2 GB of RAM. The bold numbers show the applied setting for the majority of the experiments in this work which has established a tradeoff between the elapsed time and acceptable results. The size of the extracted pixels from the target and

non-target should be summed to indicate the total amount of pixels used for training.

3.5 Other applications

As another application for the TCS, its robustness in creating a new color space for skin detection is illustrated. This usage is very practical in the field of image processing which is another motivation to think over other aspects of the new color space. The following matrix shows the obtained result by applying the new method through the linear transforma-

Fig. 13 Creating a new color space with the proposed method for skin detection.

- (a) The original image.
- (b) The converted image

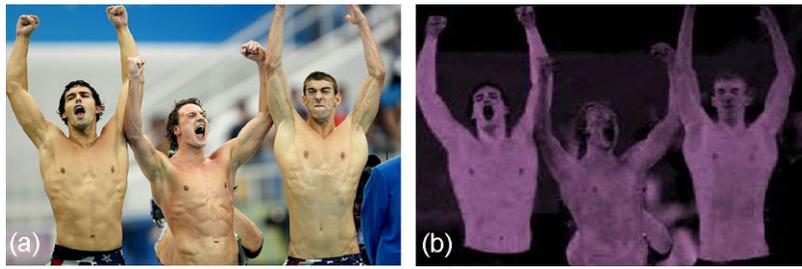


Table 3 Required time for obtaining the classified image in the TCS with different initializations

	No. of sample pixels from target and non-target								
	450 (two 15 × 15 matrices)			1250 (two 25 × 25 matrices)			2450 (two 35 × 35 matrices)		
No. of particle and max iteration	$P = 50$ $I = 100$	$P = 60$ $I = 90$	$P = 90$ $I = 120$	$P = 50$ $I = 100$	$P = 60$ $I = 90$	$P = 90$ $I = 120$	$P = 50$ $I = 100$	$P = 60$ $I = 90$	$P = 90$ $I = 120$
Total elapsed time	1'16''	1'34''	2'30''	1'46''	1'58''	4'08''	3'00''	3'09''	5'46''

tion.

$$W = \begin{bmatrix} 0.0229 & -0.1307 & -0.0126 \\ 3.1821 & 2.8361 & 2.4346 \\ -3.1594 & -2.7275 & -2.4061 \end{bmatrix}$$

Figure 13 depicts one example of skin detection in the TCS.

4 Conclusions and future work

Conventional clustering algorithms fail to solve some real-world clustering problems. A PSO-adjusted space conversion, based on quadratic conversion has been developed to deal with a particular kind of clustering problem which is the target detection for the sea pictures. In the newly generated space, patterns (points) are rearranged in such a way that the FCM can recognize intrinsic grouping of the given data. There are two innovations in this paper. Firstly, we introduced a quadratic color space conversion, and secondly we successfully tested this new color space for target detection for the sea pictures. Experimental results show that not only quadratic conversion is a new criterion for creating a more differentiator color space, but also the application in the sea target detection is a new and successful experience. All above abilities made the new color space to be efficient in more general frames. It means that our new method, unlike the related mentioned work in Sect. 1.2 that are efficient in special kind of images, or frames with special angle of view, is applicable in almost all color images and frames. For sure our work is one of the first research works on the sea target detection area that utilizes color features. One of our future works is to generalize our method to any other benchmark problems. We used just color features classification which is not appropriate in all images.

Combination of our algorithm with some other features and utilizing other training based methods like Support Vector Machine (SVM), and extending the study to a more general nonlinear case with kernel approach are other future works. In general, such a space conversion that can result in the discovery of natural grouping in data would be appreciable. Because, it is valuable in many areas like: signal representation, pattern classification/recognition, pattern understanding/analysis, and data mining. In other words, it makes it possible to work in a more sensible and more desirable space.

Appendix

The Particle swarm optimization (PSO) is a swarm intelligence-based algorithm to find a solution to an optimization problem in a search space, or model and predict social behavior in the presence of objectives. The particle swarm optimization algorithm was first developed in 1995 by James Kennedy and Russell C. Eberhart [57, 58]. The techniques have evolved greatly since then, and the original version of the algorithm is barely recognizable in the current ones. The swarm is typically modeled by particles in the multidimensional space that have a position and a velocity. These particles fly through hyperspace (i.e. R^n) and have two essential reasoning capabilities: their memory of their own best position (pbest) and knowledge of the global or their neighborhood's best (gbest). In a minimization optimization problem, problems are formulated so that "best" simply means the position with the smallest objective value. Members of a swarm communicate good positions to each other and adjust their own position and velocity based on these good positions. So a particle has the following information to make a suitable change in its position and velocity:

- A global best that is known to all and immediately updated when a new best position is found by any particle in the swarm.
- Neighborhood best that the particle obtains by communicating with a subset of the swarm.
- The local best, which is the best solution that the particle has seen locally.

In this work, the global variant of the PSO was considered because it exhibited faster convergence compared to the local one [59]. Suppose that the search space is D -dimensional, then the i th particle of the swarm can be represented by a D -dimensional vector, $X_i = (x_{i1}, x_{i2} \dots x_{iD})$. The velocity (position change) of this particle, can be represented by another D -dimensional vector $V_i = (v_{i1}, v_{i2} \dots v_{iD})$. The use of the inertia weight w has been proved to result in improved performance [60]. In a PSO system based on inertia weight, update of particles is accomplished by the following equations:

$$v_{id}^{k+1} = wv_{id}^k + c1 \times r1(xpbest_{id} - x_{id}) + c2 \times r2(xgbest_{id} - x_{id}) \quad (12)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (13)$$

where $d = 1, 2, \dots, D$; $i = 1, 2, \dots, N$, and N is the size of the swarm; $c1$ and $c2$ are positive constants; $r1$ and $r2$ are random numbers, uniformly distributed in the interval $[0, 1]$; and $k = 1, 2, \dots$, denotes the iteration number.

References

- Shi C, Xu K, Peng J, Ren L (2008) Architecture of vision enhancement system for maritime search and rescue. In: Proc. 8th International Conference on ITS Telecommunications (ITST), Oct. 2008, pp 12–17
- Eldhuset K (1996) An automatic ship and ship wake detection system for space borne SAR images in coastal regions. *IEEE Trans Geosci Remote Sens* 34(4):1010–1019
- Zhang Y, Huang WG, Zhang YG (2003) On the space remote sensing of vessels at sea with synthetic aperture radar. *Hydrogr Surv Charting* 23(1):53–57
- Jiang QS, Aitnouri EM (2000) Ship detection in radar sat SAR imagery using PNN-model. *Can J Remote Sens* 26(4):297–305
- Novak LM, Halversen SD, Owirka GJ, Hielt M (1997) Effects of polarization and resolution on SAR ATR. *IEEE Trans Aerosp Electron Syst* 33(1):102–115
- Yang W, Sun H, Xu X, Xu G (2004) Detection of ships and ship wakes in space borne SAR imagery. *Geomat Inf Sci Wuhan Univ* 29(8):682–685 (in Chinese)
- Kuo JM, Chen K-S (2003) The application of wavelets correlator for ship wake detection in SAR images. *IEEE Trans Geosci Remote Sens* 41(6):1506–1511
- Liao M, Wang C, Wang Y, Jiang L (2008) Using SAR images to detect ships from sea clutter. *IEEE Geosci Remote Sens Lett* 5(2):194–198
- Yaman C, Asari V (2007) Long-range target classification in a cluttered environment using multi-sensor image sequences. In: Proc 3rd International Conference on Recent Advances in Space Technologies (RAST), 14–16 June 2007, pp 304–308
- Yaslan Y, Giinsel B (2004) Detection of sea targets from thermal images. In: Proc the IEEE 12th Signal Processing and Communications Applications Conference, 28–30 April 2004, pp 672–675
- Yang S, He S, Lin H (2008) Video image targets detection based on the largest Lyapunov exponent. In: Proc the 9th Int. Conference for Young Computer Scientists (ICYCS), Hunan, 18–21 Nov 2008, pp 2973–2977
- Hong Z, Jiang Q, Guan H, Weng F (2007) Measuring overlap-rate in hierarchical cluster merging for image segmentation and ship detection. In: Proc Fourth Int Conf on Fuzzy Systems and Knowledge Discovery (FSKD), Haikou, 24–27 Aug 2007, pp 420–425
- Hu Y, Wu Y (2008) Number estimation of small-sized ships in remote sensing image based on cumulative projection curve. In: Proc Int Conf on Audio, Language and Image Processing (ICALIP), Shanghai, 7–9 July 2008, pp 1522–1526
- He S, Yang S, Shi A, Li T (2008) A novel image moving sea targets detection method based on the natural measure feature. In: Proc Int Symposium on Information Science and Engineering (ISISE), Shanghai, 20–22 Dec 2008, vol 2, pp 397–400
- Gonzalez RC, Woods RE (2008) Digital image processing, 3rd edn. Pearson Education/Prentice Hall, Upper Saddle River/New York
- Cirrincone G, Cirrincone M (2003) A novel self-organizing neural network for motion segmentation. *Appl Intell* 18(1):27–35
- Lin C (2007) Face detection in complicated backgrounds and different illumination conditions by using YCbCr color space and neural network. *Pattern Recognit Lett* 28:2190–2200
- Eisner A, MacLeod DIA (1980) Blue sensitive cones do not contribute to luminance. *J Opt Soc Am A* 70:121–123
- Boynton RM, Eskew RT, Olson CX (1985) Blue cones contribute to border distinctness. *Vis Res* 25(9):1349–1352
- Stockman A, MacLeod DIA, DePriest DD (1991) The temporal properties of the human short-wave photoreceptors and their associated pathways. *Vis Res* 31:189–209
- Philipp I, Rath T (2002) Improving plant discrimination in image processing by use of different color space transformations. *Comput Electron Agric* 35:1–15
- Sigal L, Sclaroff S, Athitsos V (2004) Skin color-cased video segmentation under time-varying illumination. *IEEE Trans Pattern Anal Mach Intell* 26(7):862–877
- Bahadori S, Iocchi L, Leone GR, Nardi D, Scozzafava L (2007) Real-time people localization and tracking through fixed stereo vision. *Appl Intell* 26(2):83–97
- Jones CF, Abbott AL (2004) Optimization of color conversion for face recognition. *EURASIP J Appl Signal Process* 4:522–529
- Liew AW-C, Leung SH, Lau WH (2003) Segmentation of color lip images by spatial fuzzy clustering. *IEEE Trans Fuzzy Syst* 11(4):542–549
- Diplaros A, Gevers T, Patras I (2006) Combining color and shape information for illumination-viewpoint invariant object recognition. *IEEE Trans Image Process* 15(1):1–11
- Jin L, Li D (2007) A switching vector median filter based on the CIELAB color space for color image restoration. *Signal Process* 87:1345–1354
- Brunner CC, Maristany AG, Butler DA, Vanleuween D, Funck JW (1992) An evaluation of color spaces for detecting defects in Douglas-fir veneer. *Ind Metrol* 2(3–4):169–184
- Littmann E, Ritter H (1997) Adaptive color segmentation; a comparison of neural and statistical methods. *IEEE Trans Neural Netw* 8(1):175–185
- Yang MH, Ahuja N (1998) Detecting human faces in color images. In: Proc of the International Conference on Image Processing (ICIP), Chicago, 4–7 Oct 1998, vol 1, pp 127–139
- Gabrys B, Bargiela A (2000) General fuzzy min-max neural network for clustering and classification. *IEEE Trans Neural Netw* 11(3):769–783

32. Fung G (2001) A comprehensive overview of basic clustering algorithms. Technical report, June 22, 2001
33. Coiras E, Mignotte P-Y, Petillot Y, Bell J, Lebart K (2007) Supervised target detection and classification by training on augmented reality data. *IEE Proc Radar Sonar Navig*, 1(1):83–90
34. Mirghasemi S, Banihashem E (2009) Sea target detection based on SVM method using HSV color space. In: *Proc IEEE Student Conference on Research and Development (SCoReD)*, Nov 2009, pp 555–558
35. Trinh H-H, Kim D-N, Jo K-H (2007) Supervised training database for building recognition by using cross ratio invariance and SVD-based method. *Appl Intell* 32(2):216–230
36. Power PW, Clist RS (1996) Comparison of supervised learning techniques applied to color segmentation of fruit images. In: *Proc SPIE Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Material Handling*, Boston, MA, Nov 1996, vol 2904, pp 370–381
37. Tao L-M, Peng Z-Y, Xu G-Y (2001) Features of skin color. *J Softw* 12(7):1032–1041
38. De Dios JJ, Garcia N (2003) Face detection based on a new color space YCgCr. In: *Proc International Conference on Image Processing (ICIP)*, 14–17 Sep 2003, vol 3, pp III-909-12
39. Zhao YJ, Dai SL, Xi X (2008) A Mumford-Shah level-set approach for skin segmentation using a new color space. In: *Proc Asia Simulation Conference, 7th International Conf. on Sys. Simulation and Scientific Computing (ICSC)*, 10–12 Oct 2008, pp 307–310.
40. Stehle T, Behrens A, Aach T (2008) Enhancement of visual contrast in fluorescence endoscopy. In: *Proc IEEE International Conference on Multimedia and Expo (ICME)*, Hannover, June 23–April 26, 2008, pp 537–540
41. Jia L, Liu Y (2008) A novel thresholding approach to background subtraction. In: *IEEE Workshop on Applications of Computer Vision (WACV)*, Copper Mountain, CO, 7–9 Jan 2008, pp 1–6
42. Park JB (2004) Detection of specular highlights in color images using a new color space transformation. In: *Proc IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Shenyang, 22–26 Aug 2004, pp 737–741
43. Panetta K, Qazi S, Agaian S (2008) Techniques for detection and classification of edges in color images. In: *Proc (SPIE)*, vol 6982, pp 69820W–69820W-11
44. Shen S, Szameitat AJ, Sterr A (2008) Detection of infarct lesions from single MRI modality using inconsistency between voxel intensity and spatial location—a 3-D automatic approach. *IEEE Trans Inf Technol Biomed* 12(4):532–540
45. Sentelle S, Sentelle C, Sutton MA (2002) Multiresolution-based segmentation of calcifications for the Early detection of breast cancer. *Real-Time Imaging* 8(3):237–252
46. Sun X, Qian W, Song D (2004) Ipsilateral-mammogram computer-aided detection of breast cancer. *Comput Med Imaging Graph* 28(3):151–158
47. Arfan Jaffar M, Hussain A, Mirza AM (2009) Lungs nodule detection by using fuzzy morphology from CT scan Images. In: *Proc. International Association of Computer Science and Information Technology (IACSITSC)*, Singapore, 17–20 April 2009, pp 57–61
48. Wang X-Y, Garibaldi JM, Bird B, George MW (2007) A novel fuzzy clustering algorithm for the analysis of axillary lymph node tissue sections. *Appl Intell* 27(3):237–248
49. Li H, He Y, Shen H (2007) Ship detection with the fuzzy c-mean clustering algorithm using fully polarimetric SAR. In: *Proc International Geoscience and Remote Sensing Symposium (IGARSS)*, 23–28 July 2007, pp 1151–1154
50. Hu Y, Chehdi K, Li G, Zu K (2008) An optimal road seed extraction algorithm. In: *Proc International Workshop on Education Technology and Training & International Workshop on Geoscience and Remote Sensing (ETT and GRS)*, Shanghai, 21–22 Dec 2008, vol 2, pp 771–775
51. Wang S-J, Jeng D-L, Tsai M-T (2009) Early fire detection method in video for vessels. *J Syst Softw* 82(4):656–667
52. Hu M, Zhang Q, Wang Z, Wu G (2008) An improved fuzzy c-means and Kathunen-Loeve transform method for face detection. In: *Proc the 3rd International Conference on Innovative Computing Information and Control (ICICIC)*, Dalian, Liaoning, 18–20 June 2008, pp 201
53. Rohani R, Alizadeh S, Sobhanmanesh F, Boostani R (2008) Lip segmentation in color images. In: *Proc International Conference on Innovations in Information Technology (IIT)*, 16–18 Dec 2008, pp 747–750
54. Park B-J, Pedrycz W, Oh S-K (2010) Polynomial-based radial basis function neural networks (P-RBF NNs) and their application to pattern classification. *Appl Intell* 32(1):27–46
55. Hosseini SM, Farsi H, Yazdi HS (2009) Best clustering around the color images. *Int J Comput Electr Eng* 1(1):20–24
56. Luenberger DG (1984) *Linear and non-linear programming*, 2nd edn. Addison-Wesley, Reading
57. Eberhart RC, Kennedy J (1995) A new optimizer using particle swarm theory. In: *Proc Int on Micro Machine and Human Science*, Japan, 1995, pp 39–43
58. Kennedy J, Eberhart RC (1995) Particle swarm optimization. In: *Proc IEEE Int Conf on Neural Networks*, Perth, WA, 27 Nov–01 Dec 1995, vol 4. pp 1942–1948
59. Kandil MM, Mohamed FA, Saleh F, Fayek M (2005) A new approach for optimizing back propagation training with variable gain using PSO. In: *Proc GVIP Conf, CICC*, Cairo, Egypt, 2005
60. Shi Y, Eberhart RC (1998) A modified particle swarm optimizer. In: *Proc Int Conf on Evolutionary Computation*, Anchorage, AK, 4–9 May 1998, pp 69–73



Saeed Mirghasemi received his B.Sc Degree in Electrical Engineering from Semnan University in 2006, and his M.Sc Degrees in Electrical Engineering from Malek Ashtar University of Tehran, Iran in 2009. He is currently with Islamic Azad University Parand branch as a lecturer. He has won several research grants, and has several published journal and conference papers in the field of intelligent image and signal processing. His research interests include pattern recognition based on supervised

classification methods, fuzzy classification and computational search methods, also adaptive filtering and digital circuit design. Email: s.mirghasemi@gmail.com.



Hadi Sadoghi Yazdi received the BS Degree in Electrical Engineering from Ferdowsi Mashad University of Iran in 1994, and then he received to the MS and PhD Degrees in Electrical Engineering from Tarbiat Modarres University of Iran, Tehran, in 1996 and 2005, respectively. He works in the Department of Computer Engineering as Associate Professor at Ferdowsi University of Mashhad. His research interests include pattern recognition, and optimisation in signal processing.



Mojtaba Lotfizad received the BS Degree in Electrical Engineering from Amir Kabir University, Iran, in 1980, and the MS and PhD Degrees from the University of Wales, UK, in 1985 and 1988, respectively. He then joined the Engineering Faculty Tarbiat Modarres University, Iran. He has also been a Consultant to several industrial and government organisations. His current research interests are signal processing, adaptive filtering, speech processing, and specialized processors.