Multi-scale foreground extraction on graph cut

Jiayi Liu[1], and *Kun* He[2,*]

[1]School of Software Engineering, Sichuan University, 610065 Chengdu, China
[2]School of Computer Science, Sichuan University, 610065 Chengdu, China

Abstract. In order to improve Grab Cut implementation effect for real images, we propose a novel improvement which extends the Grab Cut in three aspects: 1) a series of edge-preserved components are generated via the TV smoothing model; 2) the number of sub-regions is estimated by histogram shape analysis to remove the negative effects on the unreasonable number of the sub-regions; 3) a segmentation termination condition is constructed by integrating the multi-scale components. The experiment result indicates that this method performs well compared to other methods based on graph cut and is insensitive to sub-regions.

1 Introduction

The foreground, which plays the key role in image analysis and comprehension, is a region with semantic meanings in an image. Existed approaches to foreground extraction are based on low-level or mid-level characteristics of the image, such as edge information [1], appearance [2] and texture features [3]. However, those methods still cannot achieve correct segmentation result because of the ambiguity of those characteristics.

The Grab Cut [4] well extracted foreground with minimal user interaction via combining edge and appearance models. The appearance model is statistically described using the Gaussian Mixture Models (GMMs), in which one problem is that the number of Gaussians has a significant influence on accuracy of the segmentation [5]; the other is that the GMMs is inefficient on the inhomogeneous sub-regions. The implementation effect is low for high-textured images.

To optimize the foreground extraction effect, a novel improvement is proposed. This improvement extends the Grab Cut in three aspects. One is that a series of smoothing components are generated via the TV smoothing model. These components satisfy edge-preserved and region-smoothed. These characteristics are favored to model the inhomogeneous sub-regions via the GMMs. The other is that the number of sub-regions is estimated by histogram shape analysis to remove the negative effects on the unreasonable number of the sub-regions. At last, integrating the multi-scale appearances of an image, a segmentation termination condition is constructed using the IOU of the foreground objects.

In the next section, relevant segmentation algorithms derived from the graph cut will be reviewed. The following section proposes the non-linear smoothing model to optimize the appearance for segmentation, and analyzes the characteristics of smoothing components.

[*] Corresponding author: hekun@scu.edu.cn

Next, the GMMs is optimized using the histogram shape analysis. Finally, the experimental results and conclusion are given.

2 Previous Work

Users may be interested in different foreground regions for an image. To customize user's demands, foreground extraction approaches usually exploit information of user interaction. However, the former's extraction effect is unsatisfactory for images that the distribution of foreground and background exists a considerable overlap. The latter loses its effectiveness on high textured images.

Foreground extraction methods on graph cut, combining edges and appearance models, successfully extract foreground by the min-cut algorithm [6]. The appearance models can be divided as the given and estimation modals, note that the given modals [7, 8] is intensity distribution of foreground and background. However, this modal highly relies on the known foreground and background by the user's mark, and requires a large amount of the user interaction. The estimation modals [4, 7, 9] require user to define initial boundary box, and an image is divided into the background and foreground. Each region is statistically estimated via the GMMs in the segmentation process.

The fixed number of Gaussian in each GMMs has negative effects on segmentation. To remove the effect of unsuitable number of Gaussian, GMMs is optimized [7] by estimating the optimal number of Gaussians in each GMMs via the CLUSTER algorithm [10]. Since it does not exploit the intensity distribution of an image, it is not effective for an image with inhomogeneous sub-regions. To best model the inhomogeneous sub-regions using GMMs, the Supercut [9] implements the local similarity constraint to optimize GMMs, which guarantees that homogeneous pixels are classified into the same sub-class.

3 Image Multi-scale Analysis

Due to exploiting the given-scale edge and appearance information, foreground extraction methods on graph cut have an unsatisfactory segmentation effect on real images. We use the TV smoothing operator [11] to generate a serious of smoothed components $\boldsymbol{u}^{(i)}$ of an image $\boldsymbol{u}$. The component $\boldsymbol{u}^{(i)}\ i=0,\ 1,\ \cdots$ can be described as $\boldsymbol{u}_0=\boldsymbol{u}^{(0)}\xrightarrow{u_0}\boldsymbol{u}^{(1)}\xrightarrow{u_0}\cdots\xrightarrow{u_0}\boldsymbol{u}^{(k-1)}\xrightarrow{u_0}\boldsymbol{u}^{(k)}\xrightarrow{u_0}\cdots$

The component $\boldsymbol{u}^{(k)}$ contingents on $\boldsymbol{u}^{(k-1)}$, which smears the fine information by the smoothing operators. With the k increases, the component is getting coarser, and the intensity distribution of regions becomes tight cohesion. It is favored to model the foreground and background using GMMs. The other, the involvement of $\boldsymbol{u}_0$ in iterations simultaneously pays equal attention to raw images which contains the intrinsic singularities features, through which the edges are preserved.

4 Foreground Extraction Model

Given the boundary box around the desired object, an image $\boldsymbol{u}$ with N pixels is divided into two parts—the foreground region T_F and a background T_B . Combining edge and appearance model, an energy function is defined whose minimum will identify with a good segmentation.

$$E(\boldsymbol{\alpha},\boldsymbol{\omega},\boldsymbol{u})=R(\boldsymbol{\alpha},\boldsymbol{\omega},\boldsymbol{u})+V(\boldsymbol{u}) \tag{1}$$

Here, $\boldsymbol{\alpha} \in \{0,1\}$ is the segmentation label of each pixel, where 0 and 1 correspond to the mixture region and the background, respectively.

$$V(\boldsymbol{u}) = \sum_{j \in \Lambda_0} \gamma dis(i,j)^{-1} \exp(-\beta(u_i - u_j)^2) \tag{2}$$

The term $V(\bullet)$, denotes edge information, is expressed as a penalty weight which is high with the low gradient and low with the high gradient .To encourage smoothness, the constant γ relaxes the tendency to smoothness in regions of high contrast. In our experiments, γ is fixed to 50. Λ_0 is a set of the 8-neighboring pixels, and factor $dis(\bullet)$ is the Euclidean distance. To ensure the exponential term in (2) could switch appropriately between high and low contrast, the constant is chosen to be:

$$\beta = 0.5\left\langle (u_i - u_j)^2 \right\rangle^{-1} \ \beta = 0.5\left\langle (u_i - u_j)^2 \right\rangle^{-1} \tag{3}$$

Here $\langle \bullet \rangle$ denotes the expectation over an image sample.

The parameter $\boldsymbol{\omega}$ represents the parameters of the appearance models. The term $R(\bullet)$ evaluates the fitness of the segmentation $\boldsymbol{\alpha}$ to an image $\boldsymbol{u}$, given the models $\boldsymbol{\omega}$.

$$R(\boldsymbol{\alpha},\boldsymbol{\omega},\boldsymbol{u}) = \sum_{i \in T_F} -\log \sum_{m=1}^{K_\alpha} \eta(\boldsymbol{\alpha},m) g(\boldsymbol{\alpha},\boldsymbol{\mu}_m(k),\boldsymbol{\Sigma}_m(k),u_i) \tag{4}$$

Here, $\boldsymbol{\mu}_m(k)$ and $\boldsymbol{\Sigma}_m(k)$ are the mean vector and the covariance matrix of the m-th GMM, $K_\alpha, \alpha = 0,1$ denote the number of Gaussian in the foreground and background, respectively.

Combining edge and region information of the component, the energy function of segmentation be formulated and can be written as follows:

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \left\{ \min_{\boldsymbol{\omega}} E(\boldsymbol{\alpha},\boldsymbol{\omega},\boldsymbol{u}) \right\} \tag{5}$$

Minimization is done by using a standard minimum cut [6]

4.1 Clustering via Histogram Shape Analysis

The histogram can describe grey-level distributions of an image and the difference among sub-regions. Each sub-region corresponds to peak in the histogram. And its shape illustrates the fact that the gap between the two peaks probably result from border pixels between objects and background. Thus, the number of objects in an image can be estimated via histogram shape analysis.

If the histogram of an image is multi-modal, that is the image is high-texture or complex, it typically has many local minima and maxima. To weaken the effect of the local minima and maxima on the shape analysis, median filter is used to get the histogram pre-processed to preserve the valley. Let s denotes the grey levels of an image, $h(s):\{0,1,\cdots,255\} \to [0,1]$ denotes the histogram, and $\tilde{h}(s)$ is a smoothed histogram after the median filtering. A series of valleys $\{v_0,\cdots,v_{m-1},v_m,\cdots,v_k\}$ are detected by the signal change of the differential of $\tilde{h}(s)$, with $v_0 = \min\{s\}$ and $v_k = \max\{s\}$. These valleys separate the grey levels of an image into k intervals. The pixels of grey-level in $[v_{m-1},v_m]$ comprise the m-th subclass, and the area ration of its subclass is defined as:

$$\eta(m) = \int_{v_{m-1}}^{v_m} h(s)ds \tag{7}$$

Finally, image pixels are adaptively clustered into several sub-regions on its intensity distributions via histogram analysis. The median filter is usually short of an efficient decision rule on obtaining good results on different intervals in the histogram. As the sample size of the median filter increasing, the number of valleys falls slightly, which influences the partition of the sub-classes. In this paper, the sample size is determined as 17 by experiments.

4.2 Multi-scale Foreground Extraction

The foreground extraction task is to deduce the foreground object from the image *u* using the appearance model and edge. However， the appearance model and edge extracted from a single component cannot generate a complete correct segmentation result for the high-texture images. In different smoothing components, the inhomogeneous sub-regions are smoothed and are of compact distribution, leading that the number of Gaussians and parameters for every GMM are not the same. This difference cause that the segmentation results are inconsistency for each component. In order to evaluate segmentation performance, we use the IOU of the foreground object to measure the significant level of segmentation, which is defined as the following:

$$Ls(k)=1-\frac{F(T^{(k)})\cap F(T^{(k-1)})}{F(T^{(k)})\cup F(T^{(k-1)})} \tag{8}$$

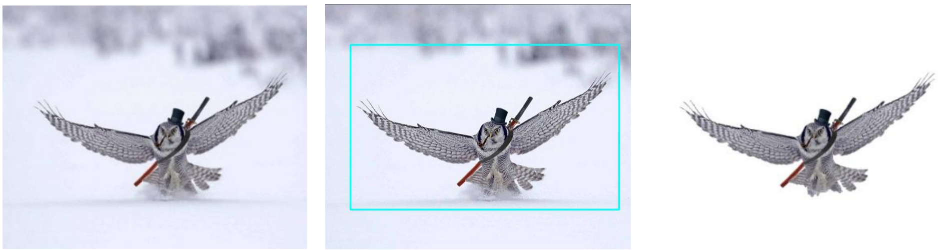
Here, $F(T)=\{i|\alpha_i=1\ and\ \alpha_i\in T, i=1,\cdots,N\}$ denotes the foreground object region; $T^{(k)}$ and $T^{(k-1)}$ represent the segmentation results of the adjacent iteration smoothing component $\boldsymbol{u}^{(k)}$ and $\boldsymbol{u}^{(k-1)}$, respectively. With the number of smoothing increasing, the significant level of segmentation monotonically decreases. According to the changes of significant level with smoothing number, the segmentation termination condition is defined as following:

$$\nabla_k Ls(k)=0 \tag{9}$$

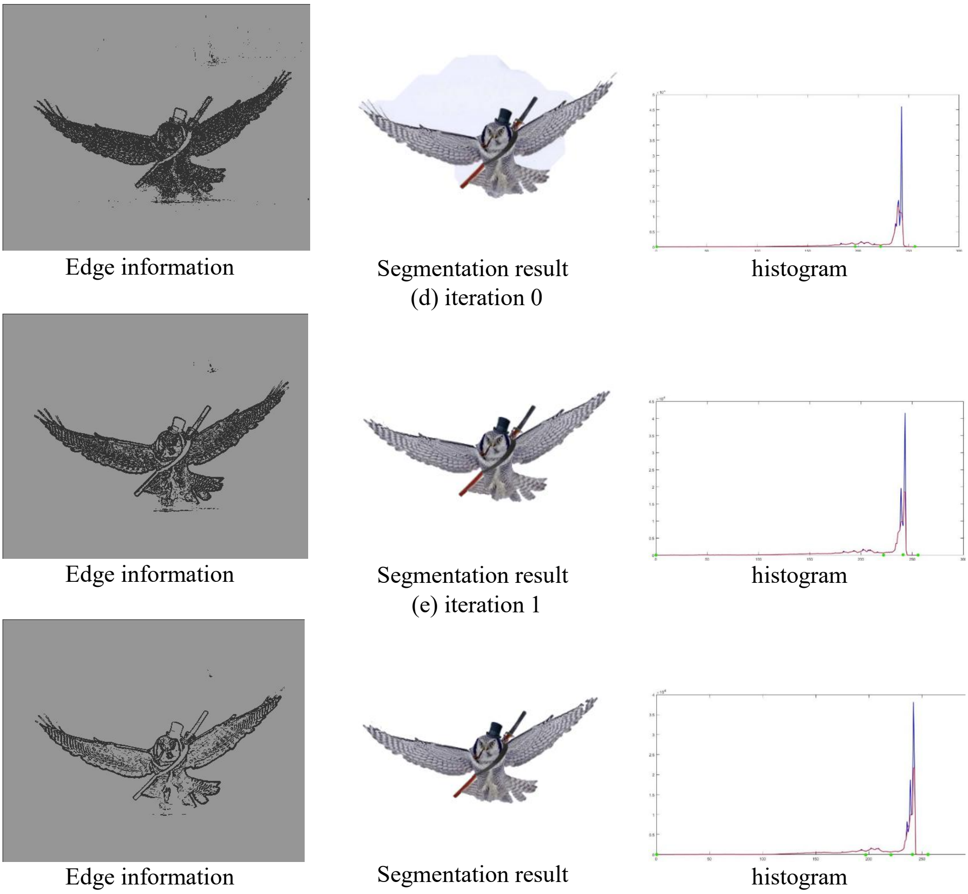
5 Experimental Results

The experiments are conducted using Visual Studio 2013 on a PC with Intel-Core i5CPU @ 3.40GHz and 4GB of RAM without any particular code optimization.

Aimed at improving segmentation performance, the non-linear smoothing is exploited to optimized appearance of images and using the histogram shape analysis to evaluate the optimal number of Gaussians for GMM construction. An image with several sub-regions (seen in the Fig.1) is used as an example. These sub-regions are smoothed and a serious of edge-preserved components are generated, which firms the segmentation of the foreground from the background. Also, the histogram shape analysis effectively optimizes the estimation of the number of Gaussians.



(a) Original image (b) initial curve (c) the ground truth



Edge information

Segmentation result
(d) iteration 0

histogram

Edge information

Segmentation result
(e) iteration 1

histogram

Edge information

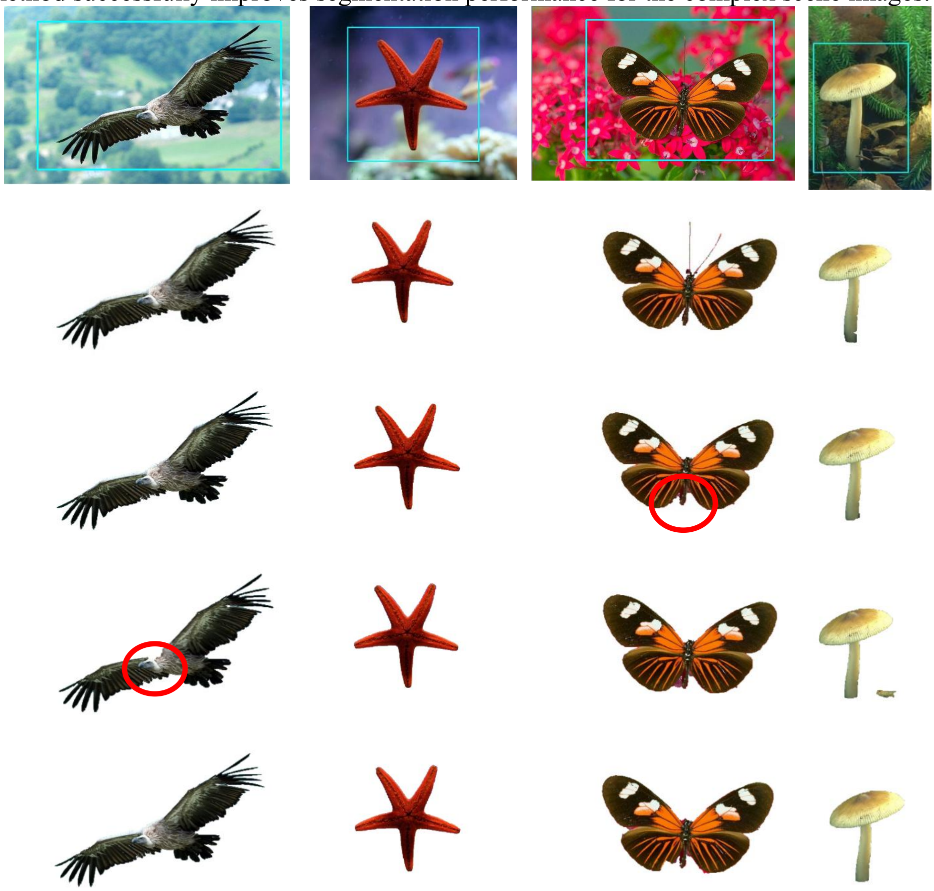
Segmentation result
(f) iteration 5

histogram

Fig. 1. The segmentation process using this method for the test image. a) the original image, b) the initial curve, c) the ground truth, (d-f) the edge information, the segmentation results, and the histogram (the blue curve denotes the histogram, the red curve is the smoothing histogram with median filter, green spots are valleys) for the component with the different times of iterations: 0,1,5.

To further test segmentation performance of this approach, experiments are carried out to compare with the correlative segmentation methods based on the graph theory, such as the Grab Cut [4] and the Grab Cut with GMM optimization [7]. The results using different methods are shown in the Fig.2. For a simple scene image with strong color contrast between the foreground and the background (i.e. the Fig.2a, b)), the effects of these three algorithms on details of edges are almost the same. And this method and the Grab Cut with GMM optimization have better performance on accurate segmentation of sub-regions than the Grab Cut (show as the red circle in Fig.2a). The reason behind it is a more reasonable estimation of color clusters for the GMM is achieved by using different methods for clustering.

For a complex scene image with multiple sub-regions (i.e. the Fig.2c, d), this method has a better result in dealing with edges comparing to the other two methods, showed as the red circle in Fig.2c. This is because in the course of smoothing, a serious of edge-preserved components are generated, which avoids the losing of edge information while smoothing sub-regions. In this situation, segmentation effect is optimized by enhancing the consistency of sub-regions and preserving edge information at the same time. Integrating the multi-scale appearances of an image and optimizing the GMMs for each region, this method successfully improves segmentation performance for the complex scene images.



(a)Bird (b) Starfish (c) Butterfly (d) Mushroom

Fig. 2. Comparison of this method with the Grab Cut, the Grab Cut with GMM optimization on the simple scene images. Row1 original images and initial curve, Row2 the ground truth, Row3 this method, Row4 the Grab Cut, Row 5 the Grab Cut with GMM optimization.

The IOU metric [12] is adopted to estimate the effectiveness of comparing methods above. It is estimated by the following:

$$\text{IOU} = \frac{F(s) \cap F(g)}{F(s) \cup F(g)} \tag{10}$$

where $F(s)$ and $F(g)$ mean the foreground object of the segmentation and ground truth, respectively. The F-measure is also used for the evaluations, it is computed by the following:

$$\text{F-measure} = \frac{2 \times precision \times recall}{precision + recall} \tag{11}$$

where,

$$\text{precision} = \frac{F(s) \cap F(g)}{F(g)}, recall = \frac{F(s) \cap F(g)}{F(s)} \tag{12}$$

The CPU time and segmentation evaluations are illustrated in Table 1 for images in the Fig. 2. The IOU and F-measure of segmentation using this method is higher than that of the other methods, which shows the superiority of this method. However, this method has higher computational cost in order to achieve better segmentation effects. The surplus CPU-time is mainly spent on the iterative computation of the non-linear smoothing for images. And the number of iteration depends on the inhomogeneous degree of the sub-region.

Table 1. The Comparison Of Cpu Time And Segmentation Evaluations On Figures 2

Segmentation methods	Fig .2a	Fig .2b	Fig.2c	Fig.2d
	465×290	550×430	512×320	320×480
This method				
Precision	0.991	0.995	0.978	0.984
Recall	0.949	0.936	0.947	0.948
F-Measure	0.962	0.966	0.956	0.966
IOU	0.941	0.953	0.949	0.933
CPU-time(s)	7.626	15.123	13.284	9.989
The Grab Cut with the GMM optimization[7]				
Precision	0.987	0.995	0.969	0.982
Recall	0.949	0.935	0.944	0.944
F-Measure	0.960	0.965	0.966	0.970
IOU	0.939	0.951	0.942	0.930
CPU-time(s)	6.495	14.119	12.443	9.125
The Grab Cut [6]				
Precision	0.975	0.994	0.970	0.976
Recall	0.948	0.935	0.943	0.938
F-Measure	0.959	0.964	0.966	0.969
IOU	0.937	0.950	0.937	0.921
CPU-time(s)	5.121	12.859	10.461	8.199

6 Conclusion

This paper proposes two modifications to the Grab Cut object segmentation method to improve segmentation performance without increasing user interaction. For images with inhomogeneous sub-regions, the non-linear smoothing algorithm is shown to optimize image appearances for images with inhomogeneous sub-regions in segmentation. Plus, the histogram shape analysis assists to estimate the suitable number of Gaussians in each GMM to well model the foreground and background. Compared to the original Grab Cut and relative improved methods, the segmentation effect using this method is superior. But it tolerates expensive computations to achieve better segmentation results. To save the computational cost, we will design a method of adaptively labelling the initial curve which adjacent to the boundaries. On the other hands, the fixed sample size of the median filter might mislead the suitable number of Gaussians for each GMM. We plan to construct an algorithm to adaptively determine the sample size of the median filter, which contributes to more accurate clusters' number by the histogram shape.

Acknowledgement

This work was supported by the Sichuan Province Natural Science Foundation of China (2016JZ0014).

References

1. S.Y. Yeo， X. Xie， I. Sazonov， and P. Nithiarasu. "Segmentation of biomedical images using active contour model with robust image feature and shape prior." International Journal for Numerical Methods in Biomedical Engineering, 2014, Vol. 30, No.2, pp. 232-248.
2. P. P. Tunga, and V. Singh. "Extraction and description of tumour region from the brain MRI image using segmentation techniques." IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology IEEE, 2017, pp. 1571-1576.
3. R. Hettiarachchi, and J. F. Peters. "Voronoï region-based adaptive unsupervised color image segmentation." Pattern Recognition, 2017, 65, pp.119-135.
4. C. Rother, V. Kolmogorov, and A. Blake. "GrabCut: interactive foreground extraction using iterated graph cuts." Acm Transactions on Graphics, 2004, Vol.23, No.3, pp. 309-314.
5. Y. Y. Boykov and M. P. Jolly. "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images." Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on IEEE, 2001, vol.1, pp.105-112.
6. V. Kolmogorov, and R. Zabih. "What Energy Functions Can Be Minimized via Graph Cuts?" IEEE Transactions on Pattern Analysis & Machine Intelligence, 2004, vol.26, no.2, pp.147-159.
7. D. Chen, B. Chen, G. Mamic, C. Fookes, and S. Sridharan. "Improved Grab Cut segmentation via GMM optimization." Digital Image Computing: Techniques and Applications IEEE Computer Society, 2008, pp. 39-45.
8. M. Tang, L. Gorelick, O. Veksler, Y. Boykov. "Grab Cut in One Cut." IEEE International Conference on Computer Vision IEEE Computer Society, 2013, pp.1769-1776.
9. S. Wu, M. Nakao, and T. Matsuda. "SuperCut: Superpixel based Foreground Extraction with Loose Bounding Boxes in One Cutting." IEEE Signal Processing Letters, 2017, vol. 24, no.12, pp.1803-1807.
10. C. A. Bouman. "Cluster: An unsupervised algorithm for modeling Gaussian mixtures." Technical report, Prude University, July 2005.
11. D. G. Lowe. "Distinctive Image Features from Scale-Invariant Key points." International Journal of Computer Vision, 2004, vol.60, no. pp.91-110.
12. A. Pratondo, CK. Chui, and SH. Ong. "Robust Edge-Stop Functions for Edge-Based Active Contour Models in Medical Image Segmentation." IEEE Signal Processing Letters, 2015, vol. 23, no. 2, pp. 222-226.