IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 28, 2019, accepted March 7, 2019, date of publication March 12, 2019, date of current version April 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2904511

Multilevel Color Image Segmentation Based on GLCM and Improved Salp Swarm Algorithm

ZHIKAI XING AND HEMING JIA, (Member, IEEE)
College of Mechanical and Electrical Engineering, Northeast Forestry University, Harbin 150040, China

Corresponding author: Heming Jia (jiaheming@nefu.edu.cn)

ABSTRACT The grayscale co-occurrence matrix (GLCM) can be adapted to segment the image according to the pixels, but the segmentation effect becomes worse as the number of threshold increases. To solve this problem, we propose an improved salp swarm algorithm (LSSA) to optimize GLCM, with the novel diagonal class entropy (DCE) as the fitness function of the GLCM algorithm. At the same time, in order to increase the optimization ability of traditional SSA algorithm, Levy flight (LF) strategy should be improved. Through experiments on the LSSA algorithm of the color natural images, the satellite images, and the Berkeley images, the segmentation quality of the segmented images is evaluated by peak signal-to-noise ratio, feature similarity, probability rand index, variation of information, global consistency error, and boundary displacement error. The experimental results show that the segmentation ability of the GLCM-LSSA algorithm is superior to other comparison algorithms and has a good segmentation ability.

INDEX TERMS Color image segmentation, GLCM, salp swarm algorithm, Levy flight.

I. INTRODUCTION

Image segmentation has always been the basic work of image processing research, and is a very challenging work. Color image segmentation is mainly based on threshold segmentation [1]–[3], clustering segmentation [4]–[6], region segmentation [7]–[9] and neural network segmentation [10]–[12]. Thresholding methods involve selecting a set of thresholds using some characteristics defined from images. The concept of graylevel co-occurrence matrix (GLCM) consider the spatial correlation among the pixels of image [13], [14]. More and more attention has been paid to GLCM, and higher quality segmentation images can be obtained by using grayscale co-occurrence matrix for image segmentation. GLCM was used in many fields, Min *et al.* [15] proposed the method which extracting gray level co-occurrence matrix of the sub-blocks SAR image, then using wavelet transform to extract the norm and the average deviation as the wavelet texture feature information of sub-blocks of sub-image. Yun and Shu [16] proposed a novel ultrasound image segmentation method by spectral clustering algorithm based on the curvelet and GLCM features. The proposed technique utilized gray level co-occurrence matrix based features and a particle swarm optimization trained feedforward neural network. The improved GLCM algorithm can improve the segmentation accuracy, so in order to better improve the image segmentation accuracy of the algorithm, it has become a common method to use the optimization algorithm to find the optimal segmentation threshold of the multi-threshold algorithm [17]–[19].

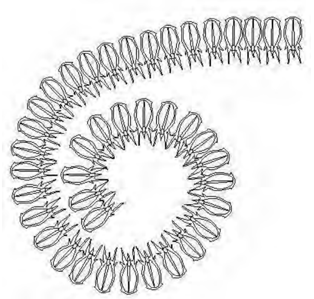


FIGURE 1. Salp chain.

Bi-level threshold image segmentation method has good segmentation ability. Many scholars study multi-threshold image segmentation method to improve the image segmentation accuracy [20]. Multi-threshold image segmentation method can effectively divide the image into multiple parts and overcome the phenomenon of similar gray value of complex images and can better find the threshold value of the image. Bhandari *et al.* [21] introduced the comparative performance study of different objective functions

The associate editor coordinating the review of this manuscript and approving it for publication was Yan-Jun Liu.

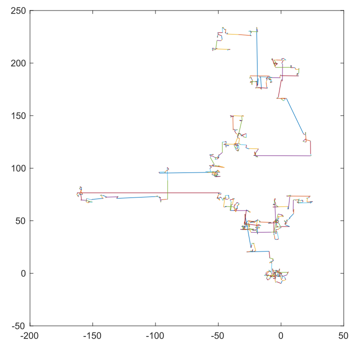


FIGURE 2. Levy's flight path.

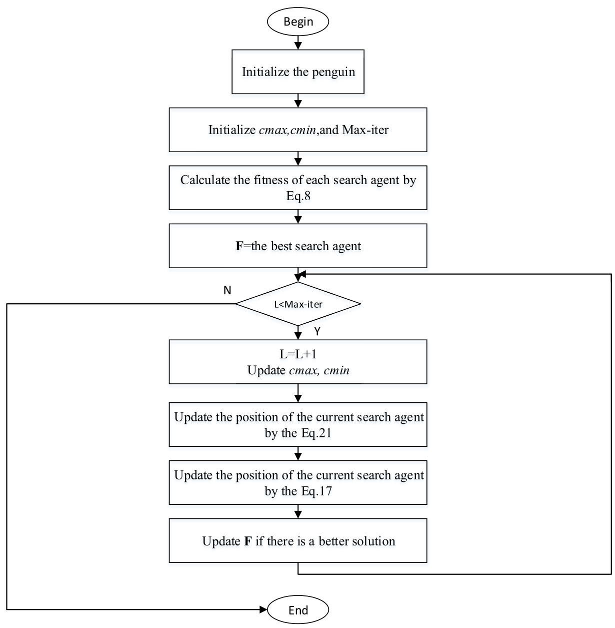


FIGURE 3. Flowchart of the GLCM-LSSA algorithm.

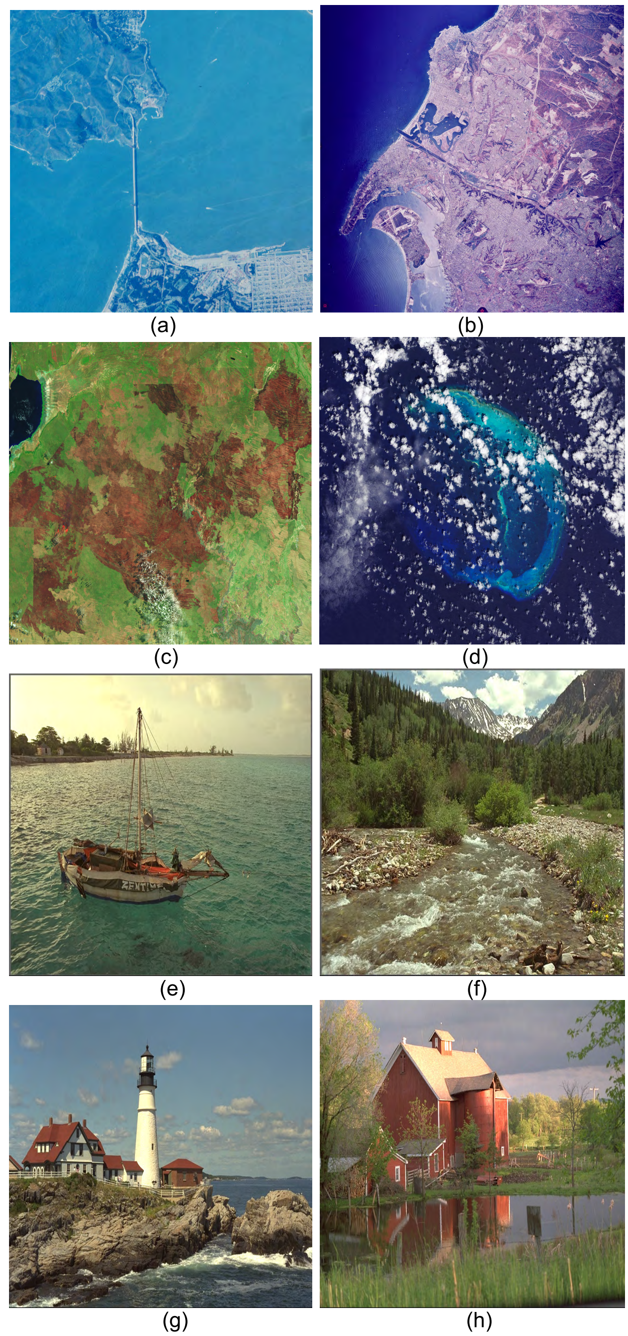


FIGURE 4. The color test images. (a) Satellite image1, (b) Satellite image2, (c) Satellite image3, (d) Satellite image4, (e) Kodim image1, (f) Kodim image2, (g) Kodim image3, (h) Kodim image4.

using cuckoo search and other optimization algorithms to solve the color image segmentation problem via multilevel thresholding. The proposed algorithm has high segmentation precision for image segmentation. Mala and Sridevi [22] proposed different methods for determining optimal thresholds using optimization techniques namely GA, PSO and hybrid model. This method solved the problem that the distribution of pixel gray was not obvious and divided the given image into a unique sub-region. Yin and Wu [23] proposed a multi-objective model which seeks to find the Pareto-optimal set with respect to Kapur and Otsu objectives.The multi-threshold Kapur image segmentation method proposed by us has better image segmentation accuracy and can better segment images. Bhandari *et al.* [24] proposed an improved ABC algorithm to optimize the image segmentation method of multi-threshold Kapur entropy. However, the time of multi-threshold Kapur entropy image segmentation algorithm was slow. Multi-threshold image segmentation method has a good image segmentation accuracy, but with the increase of the number of threshold, its segmentation accuracy will be affected. Therefore, the optimization algorithm was applied to solve the problem of threshold selection in multi-threshold image segmentation.

Intelligent optimization algorithm has attracted the attention of many scholars in recent years [25], [26]. In 2010, Iordache [27] proposed the consultant-guided search(CGS). The model of the algorithm was simple and it has good ability to solve the single and multi-dimensional mathematical functions. In 2016, Mirjalili and Lewis [28] proposed the whale optimization algorithm (WOA). Ebrahimi and Khamehchi [29] proposed the sperm whale algorithm (SWA).

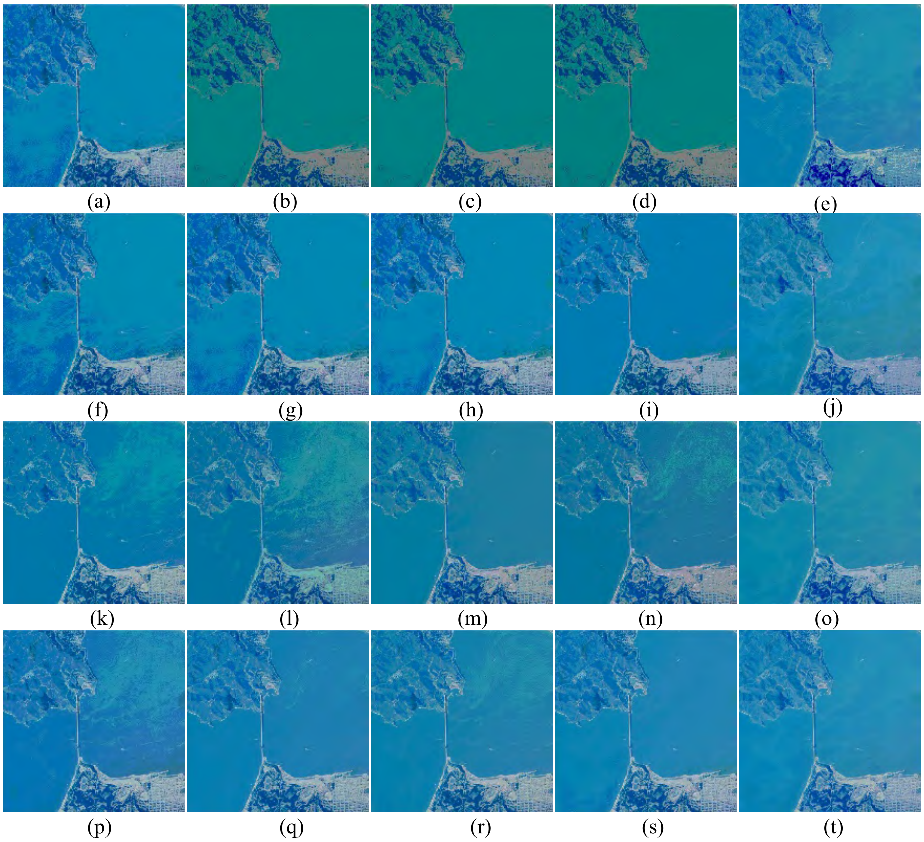


FIGURE 5. The segmentation results of satellite image1. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

TABLE 1. Parameters and references of the comparison algorithms.

Algorithm	Parameters	Value
SSA	c_2	0.5
	c_3	0.5
WOA[42]	a	[0.2]
	b	1
	l	[-1,1]
FPA[43]	P	0.5
PSO[44]	Swam size	200
	Cognitive, social acceleration	2,2
	Inertial weight	0.95-0.4
BA[22]	β	(0,1)
LSSA	Levy	1.5

Yazdani and Jolai [30] proposed the lion optimization algorithm (LOA). These algorithms have good searching ability for engineering problems and can better find the optimal value of mathematical models. In 2017, Dhiman and Kumar [31] proposed spotted hyena optimizer (SHO). The algorithm has good searching ability for the mixed mathematical function. Mirjalili *et al.* [32] proposed a novel optimization algorithm, called salp swarm algorithm (SSA), which mimiced the huddling behavior of salp swarm. The model of the algorithm was very simple, and the optimization ability was strong.

Therefore, there is no perfect optimization algorithm and the optimization algorithm should be improved to better solve engineering problems. The strategies commonly used by scholars are as follows opposition-based learning [33], Levy-flight [34] and Gaussian mutation [35]. Levy flight (LF) was a random walk strategy whose step length obeyed the Levy distribution and it could maximize the efficiency of resource searches in uncertain environments [36]. Hakli *et al.* [37] proposed the PSO algorithm which combined with Levy flight. The method could overcome the problems as being trapped in local minima due to premature convergence and weakness of global search capability. Amirsadri *et al.* [38] proposed a new algorithm benefits from simultaneously local and global search, eliminating the problem of getting stuck in local optima. The method using Levy flight improved the gray wolf optimizer (GWO). The modified algorithm balanced the exploration and exploitation of the GWO.

In this paper, Chapter 2 describes the mathematical model and principle of each basic algorithm. Chapter 3 proposes the improved GLCM-LSSA, which is improved on SSA by LF. The LSSA algorithm optimized the novel diagonal class entropy(DCE) function of GLCM. In chapter 4, standard function is carried out on the improved LSSA algorithm, and the optimization ability of the LSSA algorithm

TABLE 2. The PSNR and FSIM of each algorithm under GLCM.

T	WOA		FPA		PSO		BA		LSSA	
	PSNR	FSIM	PSNR	FSIM	PSNR	FSIM	PSNR	FSIM	PSNR	FSIM
Satellite image1										
4	17.9744	0.8654	17.5541	0.7738	16.9008	0.7734	17.8235	0.8679	**23.3737**	**0.94884**
6	23.2061	0.9463	21.3791	0.8774	21.3832	0.8814	20.9239	0.8788	**26.689**	**0.96424**
8	25.1398	0.9606	22.5334	0.9073	22.6249	0.8982	22.0579	0.8811	**29.1087**	**0.97989**
12	27.1819	0.9611	24.2467	0.9162	23.2354	0.9090	25.4246	0.9721	**32.4117**	**0.99104**
Satellite image2										
4	22.3392	0.8277	22.3470	0.8280	22.3044	0.8380	19.9078	0.7768	**26.2123**	**0.93556**
6	27.8419	0.9012	28.1530	0.8990	26.4004	0.9039	22.2960	0.8307	**29.8347**	**0.96595**
8	29.5160	0.9198	28.7455	0.9107	27.6596	0.9219	30.1602	0.9256	**31.9974**	**0.98634**
12	30.7578	0.9309	31.1217	0.9360	29.2649	0.9344	28.7392	0.9115	**32.5322**	**0.98918**
Satellite image3										
4	20.9181	0.7749	20.9579	0.7750	20.5956	0.6997	20.9783	0.7750	**26.4849**	**0.93772**
6	26.1558	0.8548	25.3810	0.8438	24.8984	0.8015	22.7276	0.8337	**26.4704**	**0.94392**
8	27.2407	0.8703	26.5908	0.8643	26.0738	0.8212	22.9602	0.8186	**31.0515**	**0.96943**
12	29.1073	0.8982	28.5093	0.8935	26.5495	0.8351	28.9372	0.8945	**33.0625**	**0.98115**
Satellite image4										
4	18.9191	0.7835	18.3870	0.7677	19.1694	0.7289	15.6841	0.6772	**26.2893**	**0.9152**
6	26.3351	0.9186	24.0158	0.8850	23.8865	0.8425	26.5919	0.9210	**28.3179**	**0.93745**
8	28.1813	0.9443	25.8145	0.9118	25.7121	0.8793	28.4094	0.9464	**30.3112**	**0.96595**
12	29.5916	0.9573	27.9701	0.9419	25.6696	0.8875	23.8150	0.8296	**30.2032**	**0.97959**
Kodim image1										
4	18.1830	0.7560	18.1572	0.7571	17.3974	0.3912	18.2168	0.7569	**21.1809**	**0.91668**
6	22.4799	0.8569	23.0034	0.8583	20.9141	0.5870	22.6567	0.8583	**25.0058**	**0.9155**
8	24.4292	0.8854	23.5906	0.8748	22.6229	0.6898	24.1356	0.8840	**27.3944**	**0.95624**
12	24.9303	0.9005	25.5315	0.9002	22.9635	0.6862	25.9704	0.9065	**29.034**	**0.97225**
Kodim image2										
4	22.4419	0.8542	21.1909	0.8160	21.0133	0.7716	22.5738	0.8542	**22.9081**	**0.85038**
6	28.0235	0.9405	25.7154	0.9060	26.3730	0.8745	22.5706	0.8094	**25.6243**	**0.89513**
8	29.3653	0.9557	27.9539	0.9342	27.0074	0.8865	25.4198	0.8494	**27.8903**	**0.89245**
12	30.4888	0.9633	29.1162	0.9418	28.0076	0.8944	30.0112	0.9362	**30.8218**	**0.96412**
Kodim image3										
4	17.2417	0.7913	16.8802	0.8247	16.9702	0.7810	16.8993	0.7749	**23.4707**	**0.92476**
6	21.7189	0.8760	21.7330	0.8973	21.8629	0.8779	21.9861	0.8757	**24.2962**	**0.94357**
8	23.2649	0.8966	26.2076	0.9377	22.2724	0.8883	24.8468	0.9526	**26.9229**	**0.93091**
12	24.2784	0.9267	27.8514	0.9616	24.9449	0.9316	26.6965	0.9639	**32.6353**	**0.98826**
Kodim image4										
4	16.6953	0.7488	16.5371	0.7478	16.6917	0.7396	16.6912	0.7398	**22.7637**	**0.84515**
6	17.1414	0.7444	16.4690	0.7612	17.8660	0.8831	17.2795	0.8600	**27.0487**	**0.89289**
8	21.5455	0.8017	21.7875	0.7891	22.1557	0.8767	19.4523	0.8895	**26.1969**	**0.90475**
12	23.5612	0.8491	22.0913	0.8012	23.0715	0.9549	22.9223	0.9369	**31.4552**	**0.9712**

is analyzed through the experimental results. In chapter 5, the GLCM-LSSA algorithm is used to segment natural color images, satellite images and Berkeley images. In order to verify the algorithm is of excellent performance in image segmentation, PSNR, FSIM, PRI, VoI, GCE, BDE and CPU time are used.

II. MATERIAL AND METHODS

A. GRAY-LEVEL CO-OCCURRENCE MATRIX (GLCM)

GLCM is a second-order statistical method that computes the frequency of pixel pairs having same gray-levels in an image and applies additional knowledge obtained using spatial pixel relations [39]. Co-occurrence matrix embeds distribution of gray-scale transitions using edge information. Since, most of the information required for computing threshold values are embedded in GLCM, it emerges as a simple yet effective technique.

Consider I as an image with 0 to L quantized gray-levels, L is considered as 256. Each matrix element of the GLCM contains the second-order statistics, probability values for changes between gray levels i and j for a particular displacement and angle. For a given distance, four angular GLCM are defined for $\theta = 0°, 45°, 90°$, and $135°$.

$$G = [g(d, 0°)+g(d, 45°)+g(d, 90°)+g(d, 135°)]/4 \quad (1)$$

where $g(\bullet)$ denotes GLCM in one direction only. Next, to prevent a negative value occurring for the entropy, we normalize

TABLE 3. The threshold levels of each algorithm under GLCM.

T	WOA			FPA			PSO		
	R	G	B	R	G	B	R	G	B
Satellite image1									
4	112 159 207 255	112 159 207 255	112 159 207 255	92 158 232 256	92 158 232 256	92 158 232 256	63 129 220 256	63 129 220 256	65 130 220 256
6	77 116 153 188 221 255	80 116 153 187 221 255	77 116 153 187 221 255	42 89 135 180 232 256	47 91 137 181 232 256	46 91 137 181 232 256	60 100 139 179 220 256	60 100 139 179 220 256	60 99 140 179 220 256
8	55 84 114 141 168 196 225 255	55 84 114 141 168 196 225 255	55 84 114 141 168 196 225 255	37 68 98 130 162 194 232 256	40 71 105 133 164 195 232 256	37 68 101 132 163 194 232 256	57 89 119 146 175 200 220 256	58 85 118 148 175 200 220 256	33 60 103 137 169 200 220 256
12	40 56 76 96 116 137 157 176 196 216 236 255	41 59 79 98 117 137 157 177 197 217 236 255	1 37 55 76 97 118 141 163 186 209 232 255	13 34 55 76 97 119 140 161 182 204 232 256	24 42 63 85 105 126 146 166 186 206 232 256	22 41 61 84 105 126 146 165 185 205 232 256	24 41 60 87 115 135 156 183 200 220 238 256	18 33 57 82 118 147 170 184 203 220 238 256	16 32 60 84 108 129 152 181 200 220 238 256
Satellite image2									
4	76 141 204 256	76 142 204 256	76 143 204 256	74 130 190 256	74 130 190 256	74 130 190 256	94 147 203 255	101 152 206 255	93 147 203 255
6	21.6668 20.3227 18.6736	74 116 155 193 224 256	74 111 147 184 212 256	20.4398 20.2744 19.058	65 101 139 178 215 256	65 101 139 178 215 256	19.3851 18.6353 18.916	46 87 128 168 210 255	46 88 130 169 210 255
8	71 98 127 154 180 204 233 256	72 98 126 155 180 204 233 256	71 98 128 158 182 204 236 256	53 80 109 138 167 196 225 256	13 56 86 120 153 186 220 256	54 82 111 140 168 196 225 256	46 79 112 144 176 207 231 255	46 78 111 144 176 207 231 255	46 75 103 132 160 188 219 255
12	22 42 63 85 108 130 150 171 192 212 236 256	22 49 70 91 113 132 153 174 193 212 236 256	23 51 71 91 112 133 154 174 193 212 236 256	13 48 68 89 109 130 151 172 193 214 234 256	13 49 70 91 111 132 152 173 194 214 234 256	13 48 68 89 109 131 151 172 193 214 234 256	45 63 82 102 121 139 158 176 194 212 233 255	46 67 85 103 121 140 158 176 194 212 233 255	46 66 84 103 121 140 158 176 194 212 233 255
Satellite image3									
4	38 75 104 144	44 64 100 142	83 134 172 193	40 71 106 152	42 74 106 144	73 120 156 187	40 70 105 151	43 76 108 147	72 119 155 187
6	25 50 66 82 99 117	27 40 56 84 91 107	52 85 120 144 167 185	26 46 63 81 101 123	18 27 61 155 164 207	41 65 93 120 142 210	26 46 63 81 102 123	32 56 75 92 112 256	46 73 103 131 152 211
8	30 37 44 55 79 94 97 172	26 46 64 81 99 129 140 160	42 60 81 91 123 143 186 216	10 14 112 121 153 208 215 237	1 12 26 43 61 77 94 164	37 56 77 101 124 142 191 214	23 39 52 67 84 103 122 183	21 40 58 72 85 99 152 174	13 42 62 86 112 135 154 213
12	29 46 51 54 73 88 97 97 121 134 134 172	19 38 45 61 78 94 104 109 133 141 157 162	48 85 104 125 148 152 166 172 178 200 207 21	1 15 29 44 56 69 85 103 123 145 173 202	18 32 47 61 74 86 98 114 132 152 178 256	21 39 58 80 103 124 140 154 167 181 196 216	16 31 44 56 69 84 99 118 136 159 180 197	17 33 49 62 76 90 106 123 140 160 177 199	19 37 51 66 86 107 127 145 160 176 193 216
Satellite image4									
4	108 142 177 199	75 113 151 179	78 109 132 153	13 21 41 53	89 121 153 183	71 101 129 156	103 136 167 195	93 126 157 185	86 120 151 231
6	73 109 119 136 165 178 186 204	80 87 92 95 129 156 177 194	51 85 98 112 127 139 144 168	51 82 102 121 141 164 185 205	39 73 95 114 135 154 174 193	40 62 78 95 114 131 148 168	51 84 108 129 149 170 189 208	66 98 130 159 186 244 253 253	48 71 89 106 121 136 152 171
8	78 87 89 105 127 143 153 168 186 208	61 91 111 126 131 151 170 173 185 197	54 84 89 99 113 132 146 147 156 166	60 78 96 113 130 148 165 180 194 210	44 66 85 101 113 128 144 159 176 196	38 61 78 93 107 120 133 145 159 175	44 45 71 89 108 128 149 170 189 208	65 85 102 118 135 152 167 182 199 249	39 62 78 93 107 122 135 149 166 190
12	77 98 102 103 112 129 152 153 173 183 196 209	73 92 102 115 131 134 134 152 160 174 199 199	56 57 72 87 104 110 114 118 130 148 159 176	20 38 167 167 193 198 219 219 221 225 235 254	1 49 69 83 95 106 123 140 156 171 186 202	39 54 68 81 94 106 117 127 137 148 161 177	33 42 57 80 99 117 134 151 167 182 195 211	1 6 33 59 81 100 119 136 153 168 183 200	26 47 64 79 93 105 117 128 138 149 162 176
Kodim image1									
4	54 99 182 195	59 101 147 207	50 90 135 194	51 91 147 212	50 82 135 206	51 81 132 205	51 91 147 213	50 82 135 206	51 81 132 205
6	36 49 56 88 135 225	47 74 99 106 146 228	45 66 70 88 112 150	35 50 70 94 119 152	42 55 71 92 118 149	40 52 64 82 108 142	34 49 69 96 122 154	42 55 72 93 119 150	41 54 67 85 110 142
8	35 38 44 58 78 113 119 225	43 58 86 91 96 119 131 214	19 47 59 62 64 77 119 225	31 40 51 68 89 113 200 233	37 46 56 69 87 109 139 237	39 50 60 74 92 117 143228	32 42 55 72 95 118 144 233	39 49 61 77 144 166 191 226	9 40 52 107 138 175 205 231
12	35 51 74 97 127 131 151 163 187 200 224 229	41 45 56 66 94 132 160 191 194 211 214 226	39 53 68 81 92 102 115 135 149 208 210 228	13 30 39 50 65 82 102 122 146 171 199 233	35 41 49 57 66 80 96 116 138 162 192 229	36 44 52 60 69 82 98 120 147 177 208 235	30 39 48 61 77 94 111 126 140 162 191 228	38 47 56 67 83 102 125 151 173 194 214 233	36 44 53 62 73 89 110 137 167 197 223 237
Kodim image2									
4	57 90 113 145	88 159 176 182	64 117 132 157	57 105 132 152	57 105 132 152	57 107 136 156	54 82 119 191	57 83 108 175	32 56 80 141
6	37 43 115 136 145 160	50 82 136 154 172 194	27 57 135 140 153 168	70 88 196 196 221 227	32 50 127 151 171 189	3 53 125 140 157 256	39 106 137 180 201 225	40 91 108 138 176 218	24 42 60 78 104 191
8	31 71 81 108 117 133 155 171	37 45 46 87 107 144 188 193	23 33 57 65 72 92 124 165	20 40 63 87 104 117 151 161	22 50 54 90 98 138 205 215	9 17 78 101 118 133 147 162	30 46 93 116 142 164 180 217	36 52 91 103 120 149 170 215	19 66 83 109 137 148 162 179
12	27 33 71 88 102 119 123 137 151 157 181 206	38 65 84 99 100 116 127 127 142 166 182 194	22 31 53 62 65 72 84 116 130 147 151 165	18 34 56 79 97 111 123 133 143 152 162 195	38 63 87 109 122 136 150 163 176 191 217 226	1 6 19 38 61 88 109 122 133 143 154 166	25 39 53 66 78 91 109 132 165 199 203 224	39 56 74 89 106 136 175 182 187 192 201 217	13 25 36 47 59 70 82 99 114 141 167 193
Kodim image3									
4	79 95 114 190	96 113 132 199	48 63 81 173	108 120 132 146	132 145 156 170	73 85 97 112	79 94 113 189	95 112 131 199	48 63 81 169
6	70 81 112 129 178 209	85 96 126 141 165 208	42 53 64 76 95 187	41 44 121 152 180 190	123 145 152 159 167 180	32 44 80 142 157 172	69 81 91 152 198 222	84 95 104 113 123 226	40 50 60 113 157 160
8	67 76 99 108 120 141 214 254	55 84 114 128 139 141 162 229	11 38 64 74 88 112 140 182	48 106 178 204 207 221 224 239	8 10 43 103 112 122 197 225	31 49 97 123 171 175 211 246	69 100 114 140 174 216 224 225	84 96 132 157 190 192 192 205	42 53 94 165 170 184 238 249
12	63 72 79 86 92 99 106 115 130 155 204 227	5 10 16 52 60 78 102 116 119 126 157 218	35 44 51 59 66 74 80 95 179 183 186 240	12 23 35 40 40 98 111 184 217 241 243 246	20 35 43 48 54 74 77 86 105 108 111 130	20 34 39 45 50 92 97 164 236 243 244 246	26 32 47 53 72 73 88 98 161 176 177 200	26 32 47 53 72 73 88 98 161 176 177 200	11 42 54 66 82 113 166 186 211 211 220 240
Kodim image4									
4	108 120 132 146	74 86 98 112	108 120 132 147	79 95 114 185	96 113 132 199	48 63 81 201	107 120 132 146	132 145 156 170	74 86 97 112
6	67 96 106 118 144 191	44 68 92 100 109 122	101 124 131 137 147 157	69 79 104 116 135 205	55 87 99 110 122 139	39 49 59 120 192 209	22 33 130 174 176 251	31 71 116 167 177 181	54 65 192 199 208 221
8	25 68 88 131 158 177 193 201	64 72 91 98 104 111 116 129	99 106 123 129 134 141 149 161	67 77 103 114 135 195 213 229	80 90 113 122 135 154 159 217	42 53 63 156 158 211 211 245	27 32 59 112 122 228 242 250	39 40 180 182 198 220 236 251	2 8 90 207 209 213 237 248
12	21 46 113 125 145 148 160 174 184 186 211 223	22 33 62 146 150 177 180 205 213 225 244 246	53 60 76 157 168 187 201 214 227 234 239 251	62 71 78 85 92 99 107 116 131 158 186 193	33 34 51 59 85 93 113 119 134 217 228 251	39 49 58 67 78 96 136 148 161 239 241 247	9 25 84 86 91 101 101 104 135 190 191 253	13 43 47 74 119 120 133 138 149 162 192 222	6 45 45 87 87 108 138 188 217 221 242 255

the final GLCM as:

$$G(i,j) = g(i,j) / \sum_{i=1}^{L} \sum_{j=1}^{L} g(i,j) \tag{2}$$

In this paper, we use the entropy feature computed from the GLCM. Let L be the number of gray levels in the image. Then the size of GLCM will be L × L. Let G(i, j) represent an element of the matrix. Then the entropy feature from the matrix is computed as

$$H = -\sum_{i=1}^{L} \sum_{j=1}^{L} G(i,j) \times \ln(G(i,j)) \tag{3}$$

TABLE 4. The optimal fitness value of each algorithm under GLCM.

T	BA			LSSA		
	R	G	B	R	G	B
Satellite image1						
4	39 88 160 226	43 84 132 205	46 84 131 205	40 80 161 227	46 89 137 208	48 86 132 206
6	17 38 60 84 110 143	28 51 73 95 120 228	3 10 61 193 196 211	18 39 111 146 191 232	27 95 120 147 182 228	29 49 69 90 113 227
8	8 23 41 110 143 182 216 239	2 6 107 160 166 205 237 240	21 36 88 107 130 157 192 233	14 32 94 119 151 188 219 242	23 79 98 118 139 164 196 233	22 39 90 111 133 162 196 233
12	7 18 31 45 60 78 98 122 154 189 219 242	16 27 40 54 68 84 101 118 137 162 196 233	15 27 39 53 68 84 101 119 140 164 195 231	12 27 44 62 81 100 120 139 164 190 217 240	19 33 47 59 74 90 107 124 144 169 201 237	21 34 47 61 74 88 103 118 134 157 188 231
Satellite image2						
4	32 50 100 129	99 138 160 184	62 93 119 144	25 49 77 167	95 127 145 186	114 147 161 175
6	98 141 157 169 180 192 209 230	35 41 69 69 70 76 179 254	24 51 74 89 103 120 136 149	21 42 56 85 132 187	68 112 127 139 148 195	9 91 138 155 166 187
8	76 113 138 153 164 173 183 195 211 230	15 65 97 118 134 146 158 169 185 213	21 40 61 75 87 98 112 127 140 151	16 28 43 56 77 119 165 201	53 92 115 127 136 145 171 201	32 39 85 129 150 161 169 193
12	16 76 86 88 122 123 125 158 181 207 223 230	53 64 85 107 125 138 149 159 169 184 206 232	2 35 51 66 77 87 96 110 126 140 152 256	20 32 45 52 57 84 90 112 114 137 187 201	77 78 94 104 110 123 133 143 146 162 174 202	32 49 61 63 106 143 158 167 168 179 196 236
Satellite image3						
4	40 71 106 152	42 74 106 144	73 120 156 187	74 104 137 169	51 106 139 167	31 46 52 103
6	26 46 101 123 148 178	18 27 164 171 176 207	41 120 142 162 184 210	25 87 101 110 129 160	43 47 93 121 158 178	22 41 58 62 92 135
8	10 121 153 157 203 208 215 237	26 43 61 77 94 114 136 164	77 101 124 142 158 174 191 214	63 90 96 113 128 143 150 177	43 77 94 111 113 129 159 185	27 35 40 57 71 103 120 163
12	1 15 29 44 56 69 85 103 123 145 173 202	18 32 47 61 74 86 98 114 132 152 178 256	21 39 58 80 103 124 140 154 167 181 196 216	38 62 84 101 116 118 120 122 140 156 176 200	45 52 53 62 91 107 112 134 146 168 181 197	17 27 41 52 57 64 72 82 100 150 154 179
Satellite image4						
4	13 21 41 53	89 121 153 183	71 101 129 156	26 54 110 170	54 92 95 149	44 76 135 167
6	51 82 102 164 185 205	39 73 95 154 174 193	40 62 114 131 148 168	26 47 72 98 112 163	53 71 102 114 158 180	45 65 76 89 139 160
8	60 78 130 148 165 180 194 210	44 101 113 128 144 159 176 196	38 93 107 120 133 145 159 175	9 24 37 55 83 123 173 208	47 64 83 116 125 134 166 196	31 50 50 62 74 107 129 174
12	20 38 167 167 193 198 219 219 221 225 235 254	1 49 69 83 95 106 123 140 156 171 186 202	39 54 68 81 94 106 117 127 137 148 161 177	17 23 30 44 77 115 144 156 207 207 207 207	3 8 25 39 56 67 67 86 116 143 166 205	15 24 50 57 71 86 116 152 158 183 187 229
Kodim image1						
4	51 91 147 212	50 82 135 206	51 81 132 205	67 103 168 189	73 135 177 210	70 94 136 168
6	35 50 119 152 189 230	42 92 118 149 185 228	40 82 108 142 182 226	59 71 99 133 182 220	75 95 125 165 185 209	41 79 95 124 140 196
8	31 40 89 113 140 170 200 233	37 46 87 109 139 171 204 237	39 50 92 117 143 171 200 228	53 87 116 136 149 185 194 235	30 72 106 124 147 150 182 218	47 85 109 129 164 173 190 194
12	13 30 39 50 65 82 102 122 146 171 199 233	35 41 49 57 66 80 96 116 138 162 192 229	36 44 52 60 69 82 98 120 147 177 208 235	20 57 85 87 106 118 123 133 171 192 205 240	57 74 94 103 126 130 160 197 198 199 224 242	48 54 59 72 87 92 120 154 176 179 204 211
Kodim image2						
4	57 105 132 152	57 105 132 152	57 107 136 156	70 101 121 177	61 109 139 168	60 102 135 174
6	70 186 196 196 221 227	32 95 127 151 171 189	82 107 125 140 157 256	24 65 101 126 128 183	43 75 114 126 147 186	38 49 90 133 142 164
8	63 87 104 117 129 140 151 161	54 90 98 138 205 206 209 215	31 51 78 101 118 133 147 162	44 76 101 112 121 167 208 220	47 57 97 97 113 133 194 208	41 64 90 109 131 148 169 188
12	18 34 56 79 97 111 123 133 143 152 162 195	38 63 87 109 122 136 150 163 176 191 217 226	1 6 19 38 61 88 109 122 133 143 154 166	24 42 58 74 94 116 119 136 145 161 169 226	37 55 69 72 101 109 118 129 147 154 180 185	25 29 45 69 87 98 119 139 143 166 186 220
Kodim image3						
4	108 120 132 146	132 145 156 170	73 85 97 112	72 100 127 178	44 83 127 155	63 138 159 216
6	41 96 121 152 180 190	123 145 152 159 167 180	32 44 80 142 157 206	68 101 126 142 225 226	44 89 111 121 139 169	57 75 108 150 155 195
8	48 106 141 177 178 204 224 239	8 10 43 43 97 103 112 225	31 49 66 66 97 123 171 246	39 51 58 95 117 145 160 168	68 83 125 125 138 156 162 207	8 62 96 117 141 154 170 173
12	12 23 35 40 40 98 111 184 217 241 243 246	20 35 43 48 54 74 77 86 105 108 111 130	20 34 39 45 50 92 97 164 236 243 244 246	57 70 87 99 103 115 130 147 153 163 174 186	11 35 60 68 79 92 117 131 143 154 165 218	34 60 84 94 97 107 128 146 150 159 178 232
Kodim image4						
4	79 95 114 185	96 113 132 199	48 63 81 201	66 91 130 183	43 66 113 190	43 81 121 170
6	69 79 104 116 135 205	55 110 122 139 178 222	39 4984 120 192 209	47 78 97 126 171 214	45 67 100 132 174 205	34 41 59 91 155 190
8	67 94 103 114 135 195 213 229	80 106 113 122 135 154 159 217	42 53 63 74 91 156 158 245	59 68 96 120 123 165 193 210	46 71 105 126 136 142 160 200	28 45 65 84 135 160 189 205
12	62 71 78 85 92 99 107 116 131 158 186 193	33 34 51 59 85 93 113 119 134 217 228 251	39 49 58 67 78 96 136 148 161 239 241 247	47 49 58 64 91 105 111 132 152 173 217 228	30 34 55 73 89 98 124 133 149 180 215 233	4 31 39 46 58 78 97 111 122 162 180 203

However, for bi-level thresholding, for a threshold value T, the DCE is computed as

$$H_A = -\sum_{i=1}^{T}\sum_{i=1}^{T} G(i,j) \times \ln(G(i,j)) \tag{4}$$

$$H_C = -\sum_{i=T+1}^{L}\sum_{i=T+1}^{L} G(i,j) \times \ln(G(i,j)) \tag{5}$$

$$H_{DCE}(T) = H_A(T) + H_C(T) \tag{6}$$

When this formulation is extended to multilevel thresholding, we consider only the diagonal regions of the

TABLE 5. Comparison of standard deviation (STD) of FSIM computed by WOA, FPA, PSO, BA and LSSA using GLCM as an objective function.

Test Images	T	WOA	FPA	PSO	BA	LSSA
Satellite image1	4	9.6048E-08	1.0350E-07	5.4459E-08	2.3562E-08	**9.4807E-16**
	6	6.9747E-08	8.7992E-08	5.0223E-08	9.0765E-09	**2.8488E-12**
	8	3.4263E-08	7.7553E-08	6.8266E-08	1.0980E-08	**5.0358E-13**
	12	6.4581E-08	9.6929E-08	7.2825E-08	8.3068E-08	**4.5393E-15**
Satellite image2	4	9.2569E-08	9.3650E-08	1.0614E-07	6.3755E-08	**4.0982E-15**
	6	3.4642E-08	8.8238E-08	1.0402E-07	8.6495E-08	**2.9022E-15**
	8	8.8983E-08	4.0697E-08	1.4491E-09	2.9907E-08	**1.5925E-15**
	12	6.0291E-08	3.5266E-08	4.5458E-08	1.8004E-05	**3.3108E-16**
Satellite image3	4	3.9561E-08	3.9770E-08	2.6169E-08	7.3730E-08	**3.5838E-10**
	6	6.4425E-09	3.7364E-08	5.5728E-02	6.1977E-08	**4.6234E-15**
	8	2.5607E-08	6.7518E-08	5.5728E-02	7.3375E-08	**2.3102E-14**
	12	1.0237E-07	7.3145E-08	5.5728E-02	2.8324E-09	**4.8963E-11**
Satellite image4	4	2.6712E-08	6.4477E-08	5.5728E-02	1.8992E-08	**4.5094E-12**
	6	3.2397E-08	1.5105E-07	9.7079E-08	9.9387E-08	**2.3793E-11**
	8	3.5267E-08	1.0079E-07	9.7229E-08	5.2707E-05	**1.5158E-15**
	12	5.6473E-08	6.4952E-08	3.1825E-03	7.5331E-08	**3.0759E-13**
Kodim image1	4	1.0176E-07	6.1097E-08	2.0845E-08	8.6422E-08	**2.2787E-15**
	6	5.0916E-08	6.2483E-08	1.0282E-07	1.0859E-07	**3.4058E-12**
	8	1.0878E-07	1.0263E-07	1.8820E-05	2.9401E-08	**2.1315E-11**
	12	7.4459E-08	6.4278E-08	4.4221E-08	6.6786E-09	**4.5382E-16**
Kodim image2	4	4.5311E-08	8.2270E-08	6.3362E-08	9.3208E-08	**3.4081E-14**
	6	8.0530E-05	5.0741E-09	1.1534E-08	2.9574E-08	**1.5080E-11**
	8	1.7406E-09	4.7931E-08	2.0822E-02	1.0584E-07	**1.8403E-12**
	12	2.0419E-08	8.9442E-08	9.4978E-08	1.2523E-08	**1.1247E-11**
Kodim image3	4	4.7905E-08	3.1897E-05	2.8021E-09	9.8165E-09	**4.1189E-10**
	6	9.1617E-08	2.2638E-02	2.2638E-02	8.2965E-05	**1.6576E-14**
	8	8.3172E-09	1.5966E-08	1.0476E-08	7.0982E-08	**1.8910E-13**
	12	6.7698E-09	8.4265E-09	7.2426E-08	9.3988E-08	**4.5658E-12**
Kodim image4	4	1.0611E-07	8.7266E-05	4.9351E-08	6.6712E-08	**3.5881E-11**
	6	5.8570E-08	7.4983E-08	1.1338E-08	3.9227E-08	**1.8768E-10**
	8	1.0070E-07	2.1627E-08	8.7760E-08	1.0782E-04	**4.4093E-13**
	12	8.1914E-08	6.1611E-08	1.0240E-07	9.5792E-08	**1.0051E-12**

TABLE 6. The calculated p-values from the Wilcoxon test for the GLCM-LSSA versus other optimizers.

Test Images	T	WOA	FPA	PSO	BA
Satellite image1	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Satellite image2	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Satellite image3	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Satellite image4	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Kodim image1	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Kodim image2	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Kodim image3	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05
Kodim image4	4	P<0.05	P<0.05	P<0.05	P<0.05
	6	P<0.05	P<0.05	P<0.05	P<0.05
	8	P<0.05	P<0.05	P<0.05	P<0.05
	12	P<0.05	P<0.05	P<0.05	P<0.05

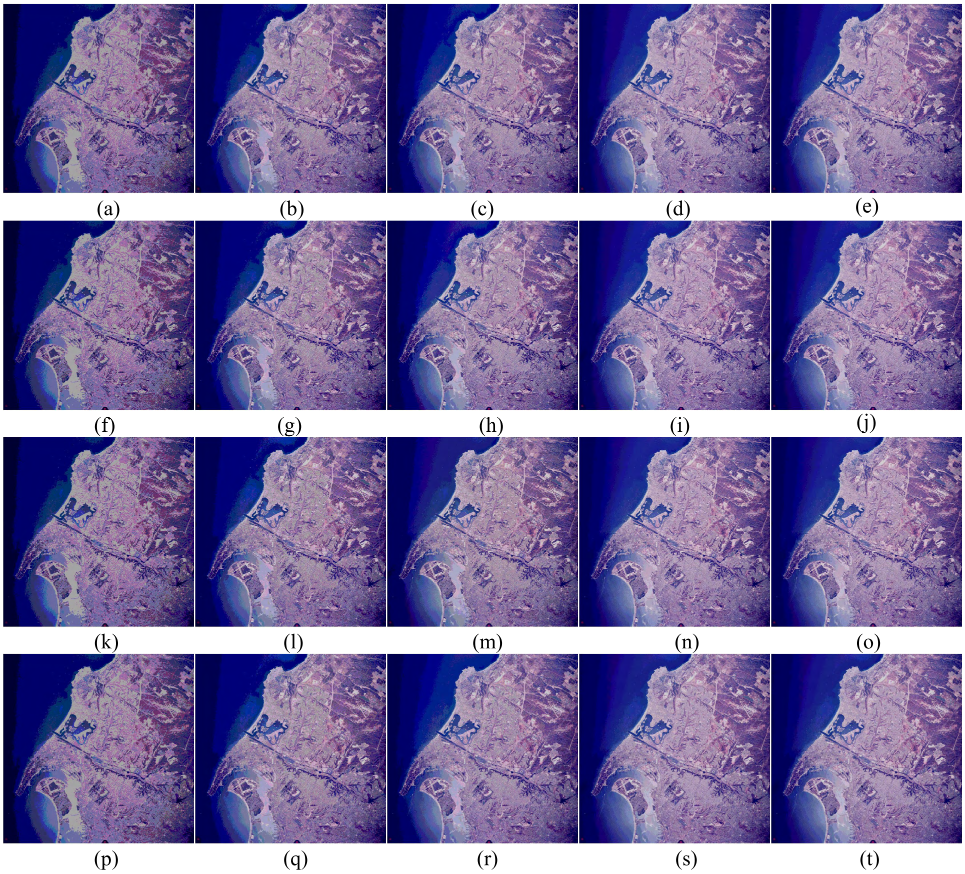


FIGURE 6. The segmentation results of satellite image2. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

GLCM for computing the DCE for each level of thresholding. The optimum thresholds are obtained when DCE is minimized. We introduce here the theoretical formulation for multilevel thresholding using DCE. For (K–1) thresholds $[T_1, T_2, \ldots, T_{K-1}]$ the DCE can be computed as

$$\begin{aligned} H_{DCE}(T_1, T_2, \ldots, T_{K-1}) = &-\sum_{i=1}^{T_1}\sum_{i=1}^{T_1} G(i,j) \times \ln(G(i,j)) \\ &-\sum_{i=T_1}^{T_2}\sum_{i=T_1}^{T_2} G(i,j) \times \ln(G(i,j)) \cdots \\ &-\sum_{i=T_{K-1}+1}^{L}\sum_{i=T_{K-1}+1}^{L} G(i,j) \\ &\times \ln(G(i,j)) \end{aligned} \tag{7}$$

The proposed objective function is:

$$\{T_1, T_2, \ldots, T_{K-1}\} = \arg\min\{H_{DCE}(T_1, T_2, \ldots, T_{K-1})\} \tag{8}$$

where, K is the number of classes.

B. KAPUR ENTROPY METHOD

Kapur's entropy method finds the optimal thresholding values by maximizing the entropy of each distinctive class or the sum of entropies based on information theory. Since it has superior performance, Kapur's entropy method have drawn the attentions of many researchers and been widely used for image segmentation problem [40].

Let there is N pixels and L gray levels in a given image, then the probability of each gray level i is the relative occurrence frequency of the gray level i, normalized by the total number of gray levels Eq.9:

$$p_i = \frac{h_i}{\sum_{i=0}^{L-1} h(i)}, \quad i = 0, \ldots, L-1 \tag{9}$$

where $h(i)$ is the number of pixels with gray level i.

For bi-level thresholding Kapur's entropy may be described by Eq.10:

$$f(t) = H_0 + H_1 \tag{10}$$

where $H_0 = -\sum_{i=0}^{t-1} \frac{p_i}{\varpi_0} \ln \frac{p_i}{\varpi_0}$, $\varpi_0 = \sum_{i=0}^{t-1} p_i$ and $H_1 = -\sum_{i=t}^{L-1} \frac{p_i}{\varpi_0} \ln \frac{p_i}{\varpi_0}$, $\varpi_1 = \sum_{i=t}^{L-1} p_i$ The optimal threshold value t^* can be found by maximizing Eq.11:

$$t^* = \arg\max(H_0 + H_1) \tag{11}$$

Further, Kapur's entropy can be easily extended for the multilevel thresholding problem as given by:

$$\begin{aligned}
H_0 &= -\sum_{i=0}^{t_1-1} \frac{p_i}{\varpi_0} \ln \frac{p_i}{\varpi_0}, \quad \varpi_0 = \sum_{i=0}^{t_1-1} p_i \\
H_1 &= -\sum_{i=t_1}^{t_2-1} \frac{p_i}{\varpi_1} \ln \frac{p_i}{\varpi_1}, \quad \varpi_1 = \sum_{i=t_1}^{t_2-1} p_i \\
H_2 &= -\sum_{i=t_2}^{t_3-1} \frac{p_i}{\varpi_2} \ln \frac{p_i}{\varpi_2}, \quad \varpi_2 = \sum_{i=t_2}^{t_3-1} p_i, \cdots \\
H_m &= -\sum_{i=t_m}^{L-1} \frac{p_i}{\varpi_m} \ln \frac{p_i}{\varpi_m}, \quad \varpi_m = \sum_{i=t_m}^{L-1} p_i
\end{aligned} \tag{12}$$

In order to search m optimal threshold values $[t_1, t_2, \cdots, t_m]$ for a given image, we try to maximize the objective function:

$$t^* = \arg\max(\sum_{i=0}^{m} H_i) \tag{13}$$

C. SALP SWARM ALGORITHM

Salps belong to the family of salpidae and have transparent barrel-shaped body. Their tissues are highly similar to jelly fishes [32]. They also move very similar to jelly fish, in which the water is pumped through body as propulsion to move forward. In deep oceans, salps often form a swarm called salp chain. This chain is illustrated in Fig.1. The main reason of this behavior is not very clear yet, but some researchers believe that this is done for achieving better locomotion using rapid coordinated changes and foraging.

To mathematically model the salp chains, the population is firstly divided into two groups: leader and followers. The leader is the salp at the front of chain, whereas the rest of salps are considered as followers. As the name of these salps implies, the leader guides swarm and the followers follow each other.

The position of salps is defined in dimensional search space where n is the number of variables of a given problem. Therefore, the position of all salps are stored in a two-dimensional matrix called x. It is also assumed that there is a food source called F in the search space as the swarm's target.

To update the position of the leader, the following equation can be represented as:

$$X_{\mathrm{j}}^1 = \begin{cases} F_j + c_1((ub_j - lb_j)c_2 + lb_j) & c_3 \geq 0 \\ F_j - c_1((ub_j - lb_j)c_2 + lb_j) & c_3 < 0 \end{cases} \tag{14}$$

where X_{j}^1 shows the position of the first salp (leader) in the jth dimension, F_j is the position of the food source in the jth dimension, ub_j indicates the upper bound of jth dimension, lb_j indicates the lower bound of jth dimension, c_1, c_2 and c_3 are random numbers. Eq.15 shows that the leader only updates its position with respect to the food source. The coefficient c_1 is the most important parameter in SSA because it balances exploration and exploitation defined as follows:

$$c_1 = 2e^{-\left(\frac{4L}{T}\right)^2} \tag{15}$$

where l is the current iteration and L is the maximum number of iterations.

The parameter c_2 and c_3 are random numbers uniformly generated in the interval of [0,1]. In fact, they dictate if the next position in jth dimension should be towards positive infinity or negative infinity as well as the step size.

To update the position of the followers, the following equations is utilized:

$$x_j^i = \frac{1}{2}at^2 + v_0 t \tag{16}$$

where i $\geq$ 2, x_j^i shows the position of ith follower salp in jth dimension, t is time, v_0 is the initial speed, and $\mathrm{a}=\frac{\mathrm{v_{final}}}{\mathrm{v_0}}$ final $\mathrm{v}=\frac{\mathrm{x\text{-}x_0}}{\mathrm{t}}$.

Because the time in optimization is iteration, the discrepancy between iterations is equal to 1, and considering $v_0 = 0$, this equation can be expressed as follows:

$$\mathrm{x_j^i} = \frac{1}{2}(\mathrm{x_j^i\text{-}x_j^{i\text{-}1}}) \tag{17}$$

With Eqs. (14) and (17), the salp chains can be simulated.

The general framework of SSA algorithm is shown as follows:

The general framework of SSA as follows:

Algorithm 1 SSA

Begin

Initialize the salp $x_i (i = 1, 2, \ldots, n)$;
Initialize *cmax,cmin,* and Max-iter;
Calculate the fitness of each search agent;
F = the best search agent ;
While (l < Max-iter)
Update c
for each search agent
Update the position of the current search agent by the Eq.14 and Eq.17;
end for
Update F if there is a better solution;
l = l+1
end while
Return F

End

D. LEVY FLIGHT

Levy's flight was firstly proposed by Levy and then described in detail by Benoit Mandelbrot. In fact, Levy flight is a random step that describes the Levy distribution [41]. Numerous studies have shown that the behavior of many animals and insects are a classic feature of Levy's flight. Levy flight is a

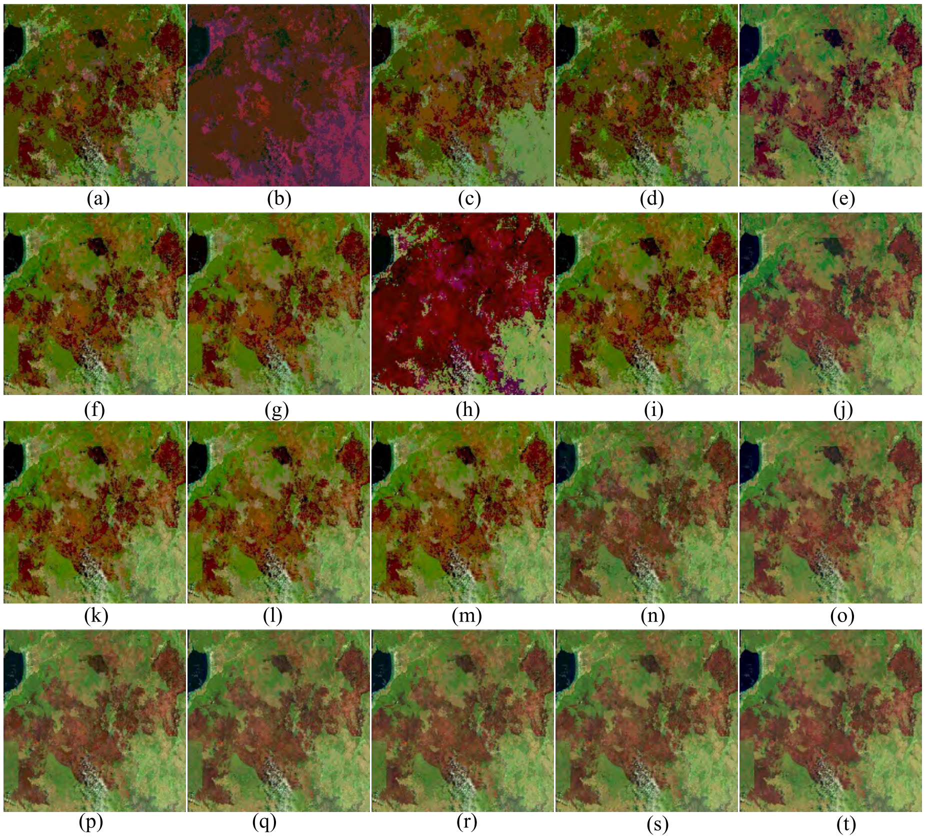


FIGURE 7. The segmentation results of satellite image3. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

special random step method, as shown in Fig.2, which is a simulation of the flight path. Its step length is always small, but occasionally it will also appear large pulsation.

The formula for Levy flight is as follows:

$$Levy \sim u = t^{-\lambda}, \quad 1 < \lambda \leq 3 \tag{18}$$

The formula for generating Levy random step proposed by Mantegna is as follows:

$$\mathrm{s} = \frac{\mu}{|v|^{1/\beta}} \tag{19}$$

where, parameter $\beta = 1.5$, μ=N(0,σ_μ^2) and v=N(0,σ_μ^2) are gamma functions.

The variance of the parameters is as follows:

$$\sigma_\mu = \left[\frac{\Gamma(1+\beta) \times \sin(\pi \times \beta/2)}{\Gamma[(1+\beta)/2] \times \beta \times 2^{(\beta-1)/2}}\right]^{1/\beta}, \quad \sigma_v = 1 \tag{20}$$

III. PROPOSED METHOD

A. IMPROVED SALP SWARM ALGORITHM (LSSA)

The SSA can solve the problem of low dimensional single mode optimization with simple and efficient solution. However, when dealing with high dimensional and complex image processing problems, traditional SSA is not very satisfactory. In order to improve the global search capability of SSA, an improved optimization algorithm of SSA is proposed in this paper. Levy flight can maximize the diversity of search domains, so that the algorithm can efficiently search the location of food sources and achieve local optimization. The Levy flight can help SSA get better optimization results, therefore to salp leader position update formula optimization, can be used to express the following mathematical formula:

$$X_{\mathrm{j}}^1 = \begin{cases} F_j + c_1((ub_j - lb_j) + lb_j)*\mathrm{Levy} & c_3 \geq 0 \\ F_j - c_1((ub_j - lb_j) + lb_j)*\mathrm{Levy} & c_3 < 0 \end{cases} \tag{21}$$

Levy flight can significantly improve the SSA's global search ability to avoid getting into local optimal values. This method not only improves the search intensity of SSA, but also improves the diversity of the algorithm. The optimization algorithm ensures that the algorithm can find the optimal value and avoid getting into local optimum, and the algorithm has better global searching ability by increasing the diversity.

B. PROPOSED GLCM-LSSA METHOD

In this section, the GLCM-LSSA is described in detail. The improved LSSA algorithm has simple structure and strong optimization ability. Therefore, the LSSA algorithm is applied to optimize the threshold selection of multi-threshold GLCM algorithm. In the GLCM-LSSA, as the fitness function of LSSA, DCE value of GLCM is used to find the

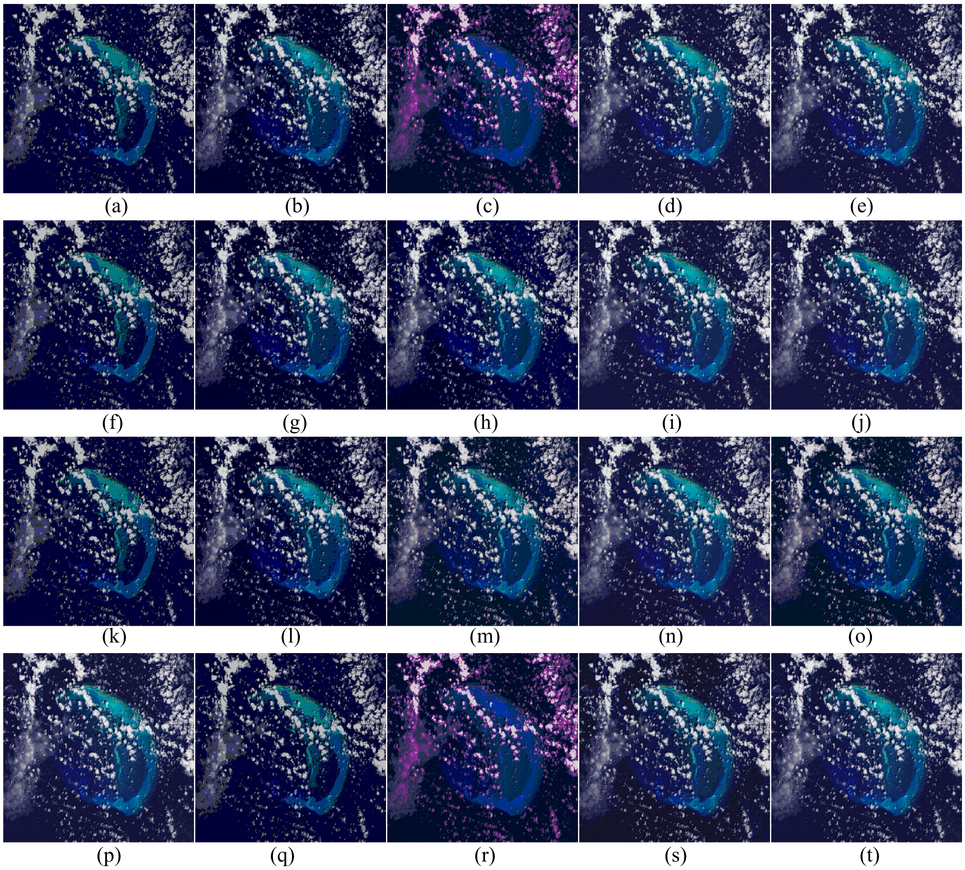


FIGURE 8. The segmentation results of satellite image4. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

minimum value of this function by LSSA, so as to find the optimal multi-threshold value of image. The image with high segmentation precision can be obtained by the optimal multi-threshold segmentation. The flowchart of the GLCM-LSSA can be seen from fig.3.

The pseudo code of the GLCM-LSSA algorithm is given below:

Algorithm 2 GLCM-LSSA

Begin

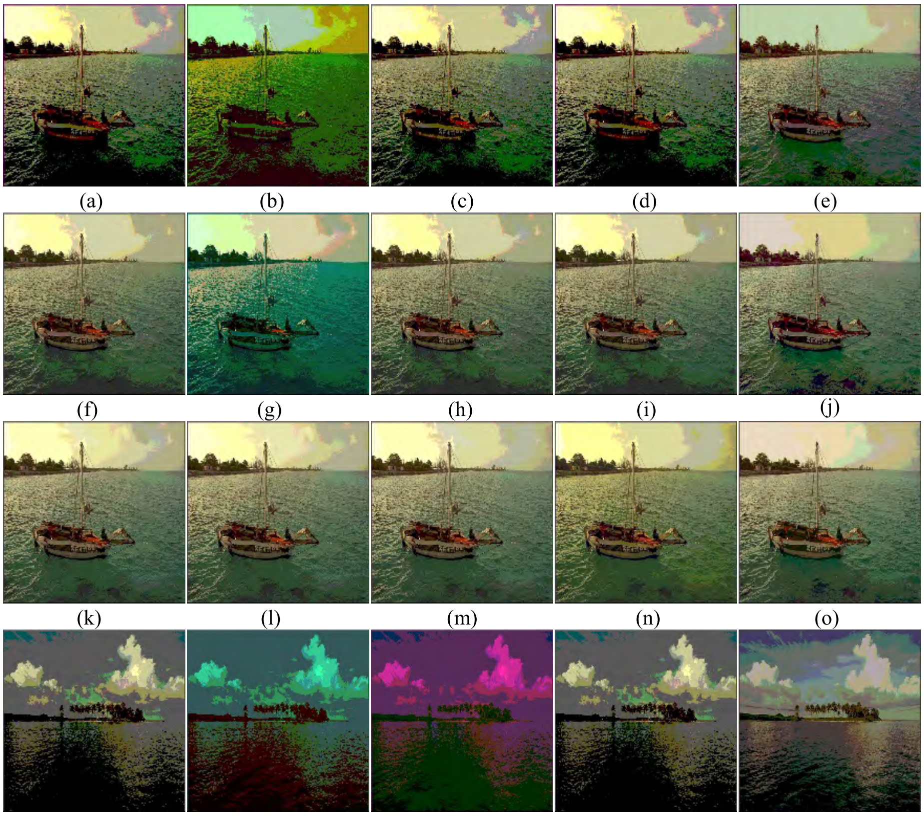
Initialize the salp $x_i (i = 1, 2, \ldots, n)$;
Initialize *cmax,cmin,* and Max-iter;
F = the best search agent by Eq.8 ;
While (l < Max-iter)
  for each search agent
    Update c
    Update the position of the current search agent by the Eq.21 and Eq.17;
  end for
  Evaluate the fitness of all salp;
  Update F if there is a better solution;
  l = l+1
end while
Return F

End

IV. EXPERIMENTS AND RESULTS

In this chapter, LSSA algorithm is applied to optimize the DCE function of GLCM algorithm. In order to better verify the image segmentation ability of GLCM-LSSA algorithm, it is compared with the optimized GLCM algorithm of WOA, PSO, FPA and BA. The color image has three color channels. In this paper, the images of the three channels are segmented, and then the three result images are fused to obtain the final segmentation result graph. Firstly, the segmentation effect and precision of GLCM-LSSA algorithm are analyzed when the threshold value is increased. Then the segmentation ability, statistical analysis and stability analysis of the proposed LSSA algorithm and other optimization algorithms in GLCM image segmentation are analyzed. Finally, the Berkeley image library is tested and analyzed. All parameters of the comparison optimization algorithm are shown in table 1.

The test images in this paper are as follows Fig. 4. The test images included color natural images and satellite images. Natural color test images (Kodim images) are accessed from http://r0k.us/graphics/kodak/. The satellite



(a) (b) (c) (d) (e)
(f) (g) (h) (i) (j)
(k) (l) (m) (n) (o)
(p) (q) (r) (s) (t)

FIGURE 9. The segmentation results of Kodim image1. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

images such as Satellite image1 and Satellite image2 has been obtained from the aerial dataset available on http://sipi.usc.edu/database/database.php?volume=aerials. Satellite image3 and Satellite image4 has been obtained from https://landsat.visibleearth.nasa.gov/. Color image segmentation requires a higher threshold level, so it is more complex to use optimization technology to solve the problem. Therefore, the optimization algorithm has the characteristics of randomness. So, all image segmentation experiments are run separately for 30 times. And the threshold levels of 4, 6, 8 and 12 are selected to find the threshold points corresponding to each color channel in the image.

The evaluation of image segmentation result graph is very important, so this paper selected PSNR and FSIM as the evaluation index of test image. The parameter of the peak signal to noise ratio (PSNR) is used to compute the peak signal to noise ratio between the original image and the segmented image [45]. The PSNR index is calculated as:

$$PSNR = 20\log(\frac{255}{RMSE})(dB) \tag{22}$$

where

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(I(i,j)-\hat{I}(i,j))^2}{M \times N}} \tag{23}$$

where, M, N is the size of the image, I is the original image, and $\hat{I}$ is the segmented image.

The feature similarity (FSIM) is used to estimate the structural similarity of the original image and the segmented image [46]. We define FSIM as:

$$FSIM = \frac{\sum_{x\in\Omega} S_L(x)\cdot PC_m(x)}{\sum_{x\in\Omega} PC_m(x)} \tag{24}$$

where Ω represents the entire image, and $S_L(x)$ indicates the similarity between the segmented images obtained through multilevel thresholding task and input image. The FSIM parameter of color RGB image is defined as:

$$FSIM = \frac{1}{O}\sum_{O} FSIM(x^O, y^O) \tag{25}$$

where, x^O and y^O represent oth channel of the original image and segmented image respectively, o is the channel number.

A. COMPARISON WITH WOA, FPA, PSO, AND BA ALGORITHM BASED MULTILEVEL SEGMENTATION TECHNIQUES

In this experiment, the results obtained by proposed GLCM based LSSA algorithm is analyzed at different threshold levels (T = 4, 6, 8, and 12) for the test images. Satellite images are difficult to be segmented because of their multimodal

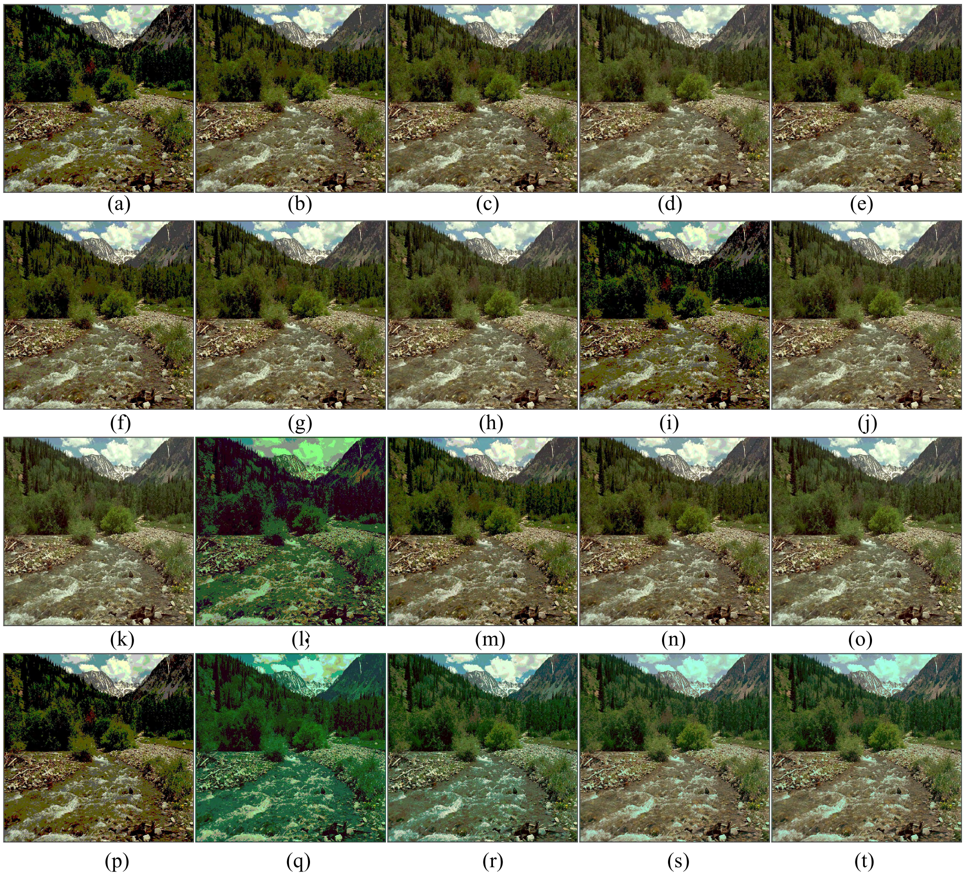


FIGURE 10. The segmentation results of Kodim image2. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

characteristics. Therefore, an algorithm based on spatial correlation is proposed to solve these problems. Table2 indicates the PSNR and FSIM values of the segmented results. Higher values of PSNR and FSIM signify better and accurate segmentation. When the number of threshold values T = 4, the PSNR value and FSIM value of each algorithm are lower. With the increase of the number of threshold values, the FSIM and PSNR values also increase, indicating that the increase of the number of threshold values can increase the segmentation precision of the image and make the segmentation result more similar to the original image.

Meanwhile, it can be clearly seen from Table 2, LSSA is better and more reliable than WOA, FPA, PSO, and BA for all the test images, because of its precise search capability, at a high threshold level (T). Performance of WOA and BA has closely followed LSSA. The solution update strategy for FPA and PSO may have led to poor results. The good results based on the LSSA algorithm are shown in table 2, and the GLCM-LSSA algorithm performs best in color images such as satellite images. The comprehensive performance ranking of the comparison algorithm is as follows: LSSA > WOA > BA > FPA > PSO. Table 3 and 4 shows the optimal threshold of the algorithm for satellite image and natural color image respectively. Therefore, LSSA has the best performance, so it determines the best threshold to produce accurate and high-quality segmentation images.

From Fig 5-12, the visual results show that this method achieves a good segmentation effect by accurately identifying the complex target and background in each level of satellite image segmentation. The image segmentation effect in Fig. 5(b, c, g) and Fig. 6(h, r) is poor, and the contour segmentation in satellite images is not clear. As the number of thresholds increases, the image segmentation quality can be enhanced from Fig. 5 and Fig 6. The LSSA algorithm in this paper has the best segmentation effect. It can be seen from Fig9-Fig.12, LSSA algorithm for natural color image segmentation effect is best, WOA and BA algorithm is essentially the same as a result, PSO algorithm segmentation results figure effect is the worst, under segmentation phenomenon exists, the target area segmentation effect is not obvious, and the existence chromatism, the best threshold segmentation results are local optimal phenomenon. Obviously, from Fig. 13 and 14, the FSIM value and PSNR value of GLCM-LSSA algorithm are better than other algorithms.

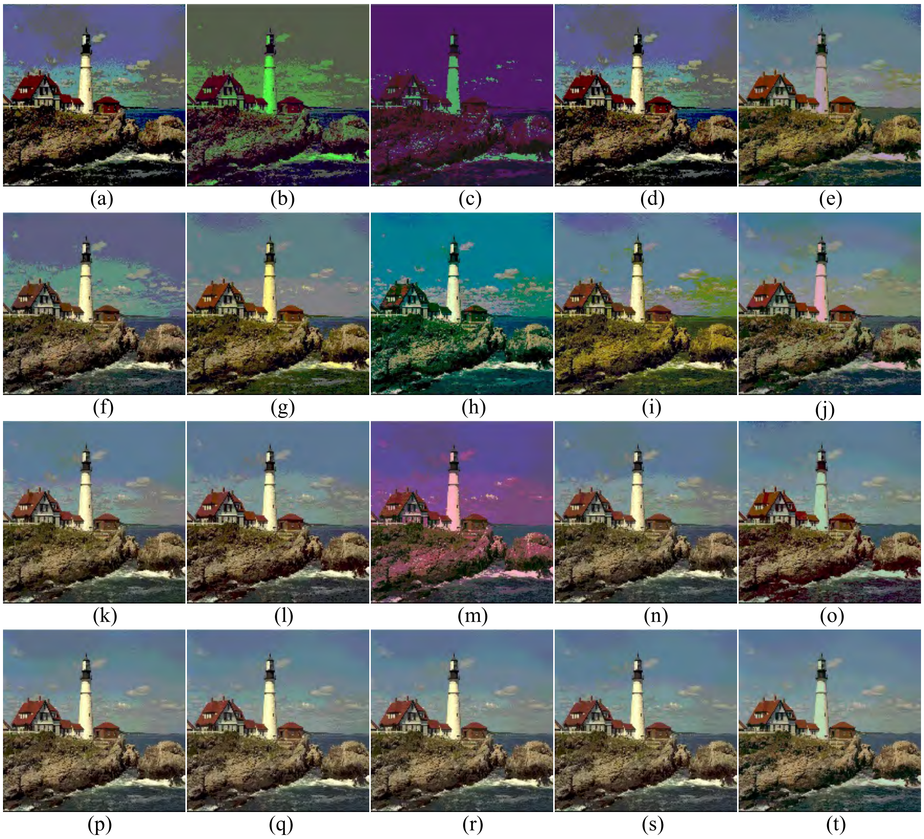


FIGURE 11. The segmentation results of Kodim image3. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

B. STABILITY AND STATISTICAL ANALYSIS

Based on the natural optimization algorithm, the results of each run are not the same. Therefore, in order to analyze the stability of the proposed algorithm based on GLCM-LSSA, we use the value of standard deviation (STD). The STD can be intuitive to the operation stability of the algorithm, and the lower the value of the algorithm, the stronger the robustness of the algorithm. Table 5 shows the STD values of each algorithm after 30 runs. It can be seen from the table that the stability of LSSA algorithm is the strongest, especially when dealing with the segmentation of satellite images, its stability is obviously better than other comparison algorithms, indicating that GLCM-LSSA algorithm has a good segmentation ability, and can find the optimal threshold of image better, more accurately and more stable.

We statistically analyze the experimental results to better observe the differences between algorithms. We use Wilcoxon rank sum test [47], a nonparametric statistical test that checks whether one of two independent samples is larger than the other. We calculate the p-value of FSIM of LSSA algorithm and WOA, FPA, PSO and BA algorithm. The experimental statistical results are shown in table 6. It can be seen from the table6 that the GLCM-LSSA algorithm is obviously better than the comparison algorithm in the statistical sense.

C. COMPARATIVE WITH MULTI-THRESHOLD KAPUR

In this section, we compare the multi-threshold GLCM algorithm with the multi-threshold Kapur algorithm, and respectively apply different optimization algorithms to optimize the two multi-threshold methods. Each multilevel image thresholding method has also been evaluated using a well-known benchmark-the Berkley segmentation data set (BSDS300) with 300 distinct images. The 300 images from the Berkeley segmentation data set (BSDS 300) available at https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/BSDS300/html/dataset/images.html. This section uses an extensive comparative study on Berkeley database by using performance metrics like Probability Rand Index (PRI), Variation of Information (VoI), Global Consistency Error (GCE), and Boundary Displacement Error (BDE) [48]–[50]. Table 7 shows the average results of PRI, BDE, GCE and VoI of ground truth results of the BSDS300 data set.

The results displayed in table 7, that the proposed technique outperforms all other compared multilevel thresholding algorithms. The GLCM-LSSA technique has obtained results close to the ground truth images. Higher values of

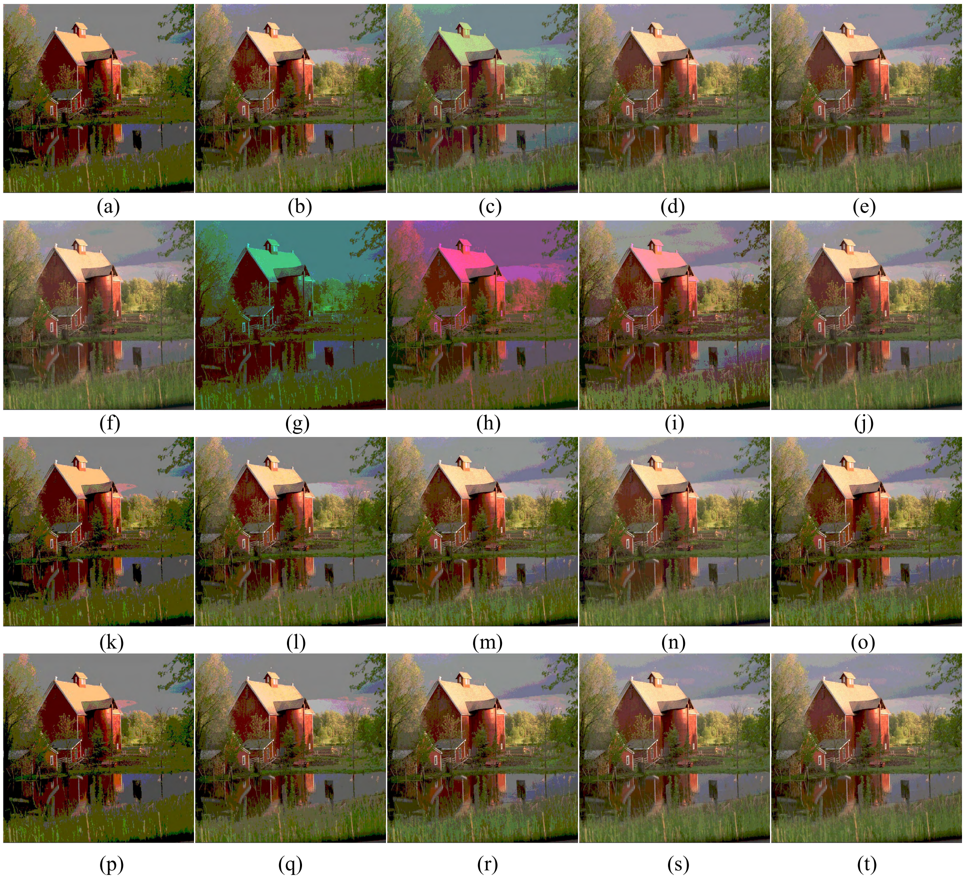


FIGURE 12. The segmentation results of Kodim image4. (a) WOA(K = 4), (b) FPA(K = 4), (c) PSO(K = 4), (d) BA(K = 4), (e) LSSA(K = 4), (f) WOA(K = 6), (g) FPA(K = 6), (h) PSO(K = 6), (i) BA(K = 6), (j) LSSA(K = 6), (k) WOA(K = 8), (l) FPA(K = 8), (m) PSO(K = 8), (n) BA(K = 8), (o) LSSA(K = 8), (p) WOA(K = 12), (q) FPA(K = 12), (r) PSO(K = 12), (s) BA(K = 12), (t) LSSA(K = 12).

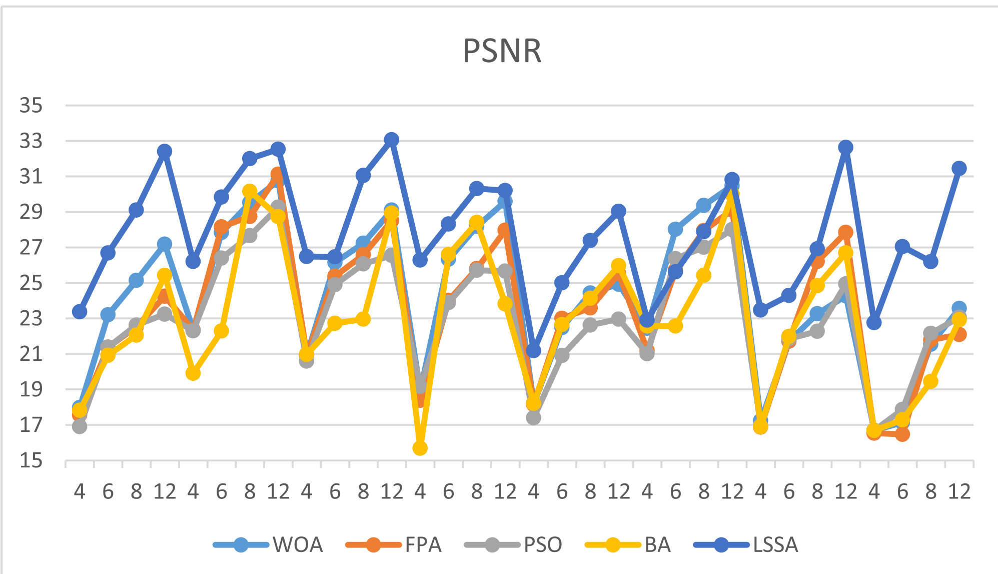


FIGURE 13. Comparison of PSNR values for different multilevel thresholding algorithms at 4, 6, 8 and 12 levels.

TABLE 7. The comparison results for the GLCM-LSSA versus other optimization image segmentation.

Algorithm	T	BDE	PRI	GCE	VOI
Ground truth		5.5862	0.9658	0.0906	1.0121
GLCM-WOA	4	10.4405	0.5522	0.3414	5.5361
	6	10.2830	0.5776	0.3950	5.9600
	8	10.9295	0.5057	0.3619	5.0236
	12	10.9178	0.5159	0.3163	5.5991
GLCM-FPA	4	11.5793	0.3997	0.3498	5.0100
	6	11.4561	0.3951	0.3268	5.7501
	8	11.9886	0.3353	0.3398	5.8606
	12	11.4242	0.3676	0.3489	5.7983
GLCM-PSO	4	11.8625	0.4976	0.4370	6.6860
	6	11.9355	0.4703	0.4816	6.9471
	8	11.7475	0.4569	0.4893	6.6230
	12	11.2957	0.4651	0.4586	6.9551
GLCM-BA	4	9.2311	0.5802	0.3805	5.8374
	6	9.9011	0.5017	0.3544	5.1052
	8	9.7801	0.5364	0.3084	5.9747
	12	9.2417	0.5788	0.3285	5.6932
GLCM-LSSA	**4**	**8.3161**	**0.7774**	**0.2586**	**3.2826**
	6	**8.0681**	**0.7077**	**0.2418**	**3.6486**
	8	**8.6627**	**0.7632**	**0.2357**	**3.4191**
	12	**8.7981**	**0.7908**	**0.2936**	**3.9171**
Kapur-WOA	4	10.1072	0.5466	0.3148	5.1813
	6	10.9604	0.5200	0.3356	5.8211
	8	10.5875	0.5087	0.3455	5.9252
	12	10.8014	0.5251	0.3049	5.3538
Kapur -FPA	4	11.9188	0.3880	0.3872	5.0661
	6	11.6666	0.3531	0.3388	5.8144
	8	11.2051	0.3988	0.3256	5.9820
	12	11.6117	0.3707	0.3458	5.6731
Kapur -PSO	4	11.2403	0.4332	0.4880	6.0434
	6	11.7472	0.4687	0.4084	6.1638
	8	11.5120	0.4818	0.4532	6.7414
	12	11.7213	0.4243	0.4438	6.4474
Kapur -BA	4	9.1456	0.5169	0.3766	5.8932
	6	9.3504	0.5241	0.3561	5.3269
	8	9.4264	0.5210	0.3597	5.2919
	12	9.4955	0.5910	0.3838	5.9358
Kapur -LSSA	4	9.8639	0.6928	0.3867	4.5408
	6	9.8765	0.6963	0.3873	4.9228
	8	9.1897	0.6489	0.3792	4.4819
	12	9.8884	0.6051	0.3155	4.5377

TABLE 8. The comparison results for the GLCM-LSSA versus novel image segmentation method.

Algorithm	BDE	PRI	GCE	VOI	Time
Ground truth	5.5862	0.9658	0.0906	1.0121	-
RW	11.6336	0.3514	0.4113	6.6978	3.1114
LSA	12.8459	0.3566	0.5449	7.7761	2.2514
MNN	9.2216	0.5476	0.4994	4.5214	5.2141
GLCM-LSSA	**8.3561**	**0.7174**	**0.2286**	**3.2126**	**1.2214**

PRI indicate better segmentation performance. While lower values of BDE, GCE, and VoI show better segmentation. It can be seen from the table that the numerical value of GLCM-LSSA algorithm is the best, indicating that its segmentation result is the closest to groundtruth and the segmentation effect is the best. And the PSO and FPA algorithm segmentation effect is the most check. It can be seen from the table that the value of the multi-threshold Kapur algorithm optimized by LSSA algorithm is superior to other optimization algorithms, indicating that the optimization ability of LSSA algorithm is relatively excellent and it can effectively improve the segmentation accuracy of image segmentation algorithm. According to the comparison between the results of GLCM-LSSA and Kapur-LSSA algorithms, the segmentation accuracy of GLCM-LSSA algorithm is higher and its stability is better than other comparison algorithms.

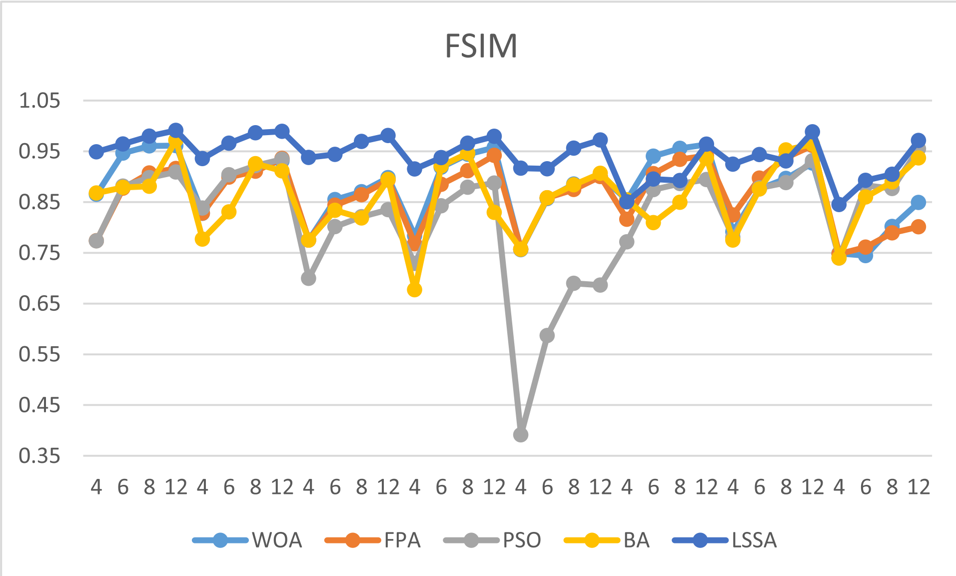


FIGURE 14. Comparison of FSIM values for different multilevel thresholding algorithms at 4, 6, 8 and 12 levels.

Therefore, GLCM-LSSA algorithm can effectively solve the problem of image segmentation.

D. COMPARATIVE WITH OTHER CLASSICAL IMAGE SEGMENTATION ALGORITHMS

In this experiment, for further showing the merits of GLCM-LSSA method, comparison is performed with other classical image segmentation algorithms, such as Random walk (RW) [51], A Level Set Approach to Image Segmentation (LSA) [52] and Multi-scale Convolutional Neural Network (MNN) [53]. Each image segmentation method has also been evaluated using a well-known benchmark-the Berkley segmentation data set (BSDS300) with 300 distinct images. Table 8 shows the average results of PRI, BDE, GCE, VoI and CPU time of ground truth results of the BSDS300 data set.

The results displayed in table 8, that the proposed technique outperforms all other compared algorithms. The GLCM-LSSA technique has obtained results close to the ground truth images. It can be seen from the table that the numerical value of GLCM-LSSA algorithm is the best, indicating that its segmentation result is the closest to groundtruth and the segmentation effect is the best. And the LSA and MNN algorithm segmentation effect is the most check. It can be seen from the CPU time of each image segmentation algorithm that the time of GLCM-LSSA algorithm is the shortest, indicating that the algorithm has good robustness. It can be seen from the analysis above, GLCM-LSSA algorithm can effectively solve the problem of image segmentation and the CPU time is short.

V. CONCLUSIONS

This paper proposes an improved salp swarm algorithm to optimize multi-threshold GLCM image segmentation method. We use Levy flight to improve the SSA algorithm, the method can balance the exploration and exploitation. The LSSA algorithm is used to optimize GLCM multi-threshold image segmentation method. In order to verify the proposed algorithm is of excellent performance in image segmentation, PSNR, FSIM, PRI, VoI, GCE, BDE and CPU time methods are used. Through the experiment and analysis of color natural image, satellite image and Berkeley image, the experiment proves that GLCM-LSSA algorithm has better image segmentation effect. And then, we compare GLCM-LSSA with Kapur-LSSA algorithm and classic image segmentation. The experimental results show that GLCM-LSSA algorithm can obtain better segmentation results and robustness are better. Therefore, GLCM-LSSA algorithm has good image segmentation ability and can better handle complex image segmentation tasks.

As a scope of further research, we will continue to study and improve the optimization ability of the LSSA algorithm to solve more complex optimization problems. Meanwhile, we will apply the GLCM-LSSA algorithm to solve more complex image segmentation problems, such as medical image segmentation and plant phenotype image segmentation.

REFERENCES

[1] J. Yang, Z. Gui, L. Zhang, and P. Zhang, "Aperture generation based on threshold segmentation for intensity-modulated radiotherapy treatment planning," *Med. Phys.*, vol. 45, no. 4, pp. 1758–1770, 2018.

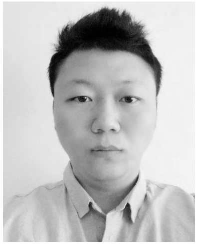
[2] L. Xu, H. Jia, C. Lang, X. Peng, and K. Sun, "A novel method for multilevel color image segmentation based on dragonfly algorithm and differential evolution," *IEEE Access*, vol. 7, pp. 19502–19538, 2019. doi: 10.1109/ACCESS.2019.2896673.

[3] A. R. J. Fredo, R. S. Abilash, and C. S. Kumar, "Segmentation and analysis of damages in composite images using multi-level threshold methods and geometrical features," *Measurement*, vol. 100, pp. 270–278, Mar. 2017.

[4] M. W. Ayech and D. Ziou, "Terahertz image segmentation using k-means clustering based on weighted feature learning and random pixel sampling," *Neurocomput.*, vol. 175, pp. 243–264, Jan. 2016.

[5] R. J. Kuo, C. H. Mei, F. E. Zulvia, and C. Y. Tsaib, "An application of a metaheuristic algorithm-based clustering ensemble method to APP customer segmentation," *Neurocomputing*, vol. 205, pp. 116–129, Sep. 2016.

[6] S. Yong, Z. Chen, Z. Qi, F. Meng, and L. Cui, "A novel clustering-based image segmentation via density peaks algorithm with mid-level feature," *Neural Comput. Appl.*, vol. 28, no. S1, pp. 29–39, 2016.
[7] I. Luengo *et al.*, "SuRVoS: Super-region volume segmentation workbench," *J. Struct. Biol.*, vol. 198, no. 1, pp. 43–53, 2017.
[8] T. Cui, J. Tian, E. Wang, and Y. Tang, "Single image dehazing by latent region-segmentation based transmission estimation and weighted L_1-norm regularization," *IET Image Process.*, vol. 11, no. 2, pp. 145–154, 2016.
[9] P. Gil and B. Alacid, "Oil spill detection in terma-side-looking airborne radar images using image features and region segmentation," *Sensors*, vol. 18, no. 1, p. 151, Jan. 2018.
[10] B. Su and S. Lu, "Accurate recognition of words in scenes without character segmentation using recurrent neural network," *Pattern Recognit.*, vol. 63, pp. 397–405, Mar. 2017.
[11] O. K. Oyedotun and A. Khashman, "Document segmentation using textural features summarization and feedforward neural network," *Appl. Intell.*, vol. 45, no. 1, pp. 198–212, 2016.
[12] J. Lei, G. Li, J. Zhang, Q. Guo, and D. Tu, "Continuous action segmentation and recognition using hybrid convolutional neural network-hidden markov model model," *IET Comput. Vis.*, vol. 10, no. 6, pp. 537–544, Sep. 2016.
[13] L. Li and Q. An, "An in-depth study of tool wear monitoring technique based on image segmentation and texture analysis," *Measurement*, vol. 79, pp. 44–52, Feb. 2016.
[14] C. Sompong and S. Wongthanavasu, "An efficient brain tumor segmentation based on cellular automata and improved tumor-cut algorithm," *Expert Syst. Appl.*, vol. 72, pp. 231–244, 2016.
[15] M. Wang, S.-D. Zhou, H. Bai, N. Ma, and S. Ye, "SAR water image segmentation based on GLCM and wavelet textures," in *Proc. Int. Conf. Wireless Commun. Netw. Mobile Comput. (WiCOM)*, Sep. 2010, pp. 1–4.
[16] T. Yun and H. Shu, "Ultrasound image segmentation by spectral clustering algorithm based on the curvelet and GLCM features," in *Proc. Int. Conf. Electr. Control Eng.*, Sep. 2011, pp. 920–923.
[17] F. R. de Siqueira, W. R. Schwartz, and H. Pedrini, "Multi-scale gray level co-occurrence matrices for texture description," *Neurocomputing*, vol. 120, no. 10, pp. 336–345, 2013.
[18] W. Gomez, W. C. A. Pereira, and A. F. C. Infantosi, "Analysis of co-occurrence texture statistics as a function of gray-level quantization for classifying breast ultrasound," *IEEE Trans. Med. Imag.*, vol. 31, no. 10, pp. 1889–1899, Oct. 2012.
[19] L. Wang, X. Tian, W. Wang, Y. Li, and L. Lei, "Cutting and extruding processing technology for ceramics based on edge-chipping effect," *Int. J. Adv. Manuf. Technol.*, vol. 84, nos. 1–4, pp. 673–678, 2016.
[20] S. Dey, S. Bhattacharyya, and U. Maulik, "New quantum inspired metaheuristic techniques for multi-level colour image thresholding," *Appl. Soft Comput.*, vol. 46, pp. 677–702, Sep. 2016.
[21] A. K. Bhandari, A.Kumar, S.Chaudhary, and G.K.Singh, "A novel color image multilevel thresholding based segmentation using nature inspired optimization algorithms," *Expert Syst. Appl.*, vol. 63, pp. 112–133, Nov. 2016.
[22] C. Mala and M. Sridevi, "Multilevel threshold selection for image segmentation using soft computing techniques," *Soft Comput.*, vol. 20, no. 5, pp. 1793–1810, 2016.
[23] P.-Y Yin and T.-H Wu, "Multi-objective and multi-level image thresholding based on dominance and diversity criteria," *Appl. Soft Comput.*, vol. 54, pp. 62–73, May 2017.
[24] A. K.Bhandari, A. Kumar, and G. K. Singh, "Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions," *Expert Syst. Appl.*, vol. 42, no. 3, pp. 1573–1601, 2015.
[25] H.Liang, H. Jia, X. Zhikai, J. Ma, and X. Peng, "Modified grasshopper algorithm based multilevel thresholding for color image segmentation," *IEEE Access*, vol. 7, pp. 11258–11295, Jan. 2019. doi: 10.1109/ACCESSS.2019.2891673.
[26] M. R. Shakarami and I. F. Davoudkhani, "Wide-area power system stabilizer design based on grey wolf optimization algorithm considering the time delay," *Elect. Power Syst. Res.*, vol. 133, pp. 149–159, Apr. 2016.
[27] S. Iordache, "Consultant-guided search: A new metaheuristic for combinatorial optimization problems," in *Proc. 12th Annu. Conf. Genet. Evol. Comput.*, 2010, pp. 225–232.
[28] S. Mirjalili and A. Lewis, "The Whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.
[29] A. Ebrahimi and E. Khamehchi, "Sperm whale algorithm: An effective metaheuristic algorithm for production optimization problems," *J. Natural Gas Sci. Eng.*, vol. 29, pp. 211–222, Feb. 2016.
[30] M. Yazdani and F. Jolai, "Lion optimization algorithm (LOA): A nature-inspired metaheuristic algorithm," *J. Comput. Des. Eng.*, vol. 3, no. 1, pp. 24–36, 2016.
[31] G. Dhiman and A. Kaur, "Spotted Hyena Optimizer for Solving Engineering Design Problems," in *Proc. Int. Conf. Mach. Learn. Data Sci. (MLDS)*, Dec. 2017, pp. 114–119.
[32] S. Mirjalili *et al.*, "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, pp. 163–191, 2017.
[33] Y. Feng *et al.*, "Opposition-based learning monarch butterfly optimization with Gaussian perturbation for large-scale 0–1 knapsack problem," *Comput. Electr. Eng.*, vol. 67, pp. 454–468, Apr. 2018.
[34] A. Palaiah *et al.*, "Clustering using cuckoo search Lévy flight," in *Proc. Int. Conf. Adv. Comput. Commun. Inform. (ICACCI)*, Sep. 2016, pp. 567–572.
[35] G. Sun, Y. Lan, and R. Zhao, "Differential evolution with Gaussian mutation and dynamic parameter adjustment," *Soft Comput.*, vol. 23, no. 5, pp. 1615–1642, Mar. 2019.
[36] A. A. Dubkov, B. Spagnolo, and V. V. Uchaikin, "Levy flight superdiffusion: An introduction," *Int. J. Bifurcation Chaos*, vol. 18, no. 9, pp. 2649–2672, 2008.
[37] H. Hakli and H. Uguz, "A novel particle swarm optimization algorithm with Levy flight," *Appl. Soft Comput.*, vol. 23, pp. 333–345, Oct. 2014.
[38] S. Amirsadri, S. J. Mousavirad, and H. Ebrahimpour-Komleh, "A Levy flight-based grey wolf optimizer combined with back-propagation algorithm for neural network training," *Neural Comput. Appl.*, vol. 30, no. 12, pp. 3707–3720, Dec. 2018.
[39] Q. Wu, Y. Gan, B. Lin, Q. Zhang, and H. Chang,"An active contour model based on fused texture features for image segmentation," *Neurocomputing*, vol. 151, pp. 1133–1141, Mar. 2015.
[40] S. Pare, A. Kumar, V. Bajaj, and G. K. Singh, "A multilevel color image segmentation technique based on cuckoo search algorithm and energy curve," *Appl. Soft Comput.*, vol. 47, pp. 76–102, Oct. 2016.
[41] R. Jensi and G. W. Jiji, "An enhanced particle swarm optimization with levy flight for global optimization," *Appl. Soft Comput.*, vol. 43, pp. 248–261, Jun. 2016.
[42] M. A. E. Aziz, A. A. Ewees, and A. E. Hassanien, "Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation," *Expert Syst. Appl.*, vol. 83, pp. 242–256, Oct. 2017.
[43] J. Yunzhi, Y. Wei-Chang, H. Zhifeng, and Y. Zhenlun, "A cooperative honey bee mating algorithm and its application in multi-threshold image segmentation," *Inf. Sci.*, vol. 369, pp. 171–183, Nov. 2016.
[44] Y. T. Lu, W. L. Zhao, and X. B. Mao, "Multi-threshold image segmentation based on improved particle swarm optimization and maximum entropy method," *Adv. Mater. Res.*, vol. 989–994, pp. 3649–3653, Jul. 2014.
[45] F. A. Fardo, V. H. Conforto, F. C. de Oliveira, and P. S. Rodrigues, *A Formal Evaluation of PSNR as Quality Measurement Parameter for Image Segmentation Algorithms*. Sao Paulo, Brazil: FEI Univ. Center, 2016.
[46] S. Pare, A. K. Bhandari, A. Kumar, and G. K. Singh, "An optimal color image multilevel thresholding technique using grey-level co-occurrence matrix," *Expert Syst. Appl.*, vol. 87, pp. 335–362, Nov. 2017.
[47] D. Kosiorowski, J. P. Rydlewski, and M. Snarska, "Detecting a structural change in functional time series using local Wilcoxon statistic," in *Statistical Papers*. Berlin, Germany: Springer-Verlag, 2017. doi: 10.1007/s00362-017-0891-y.
[48] H. Gao, Y. Tang, L. Jing, H. Li, and H. Ding, "A novel unsupervised segmentation quality evaluation method for remote sensing images," *Sensors*, vol. 17, no. 10, p. 2427, Oct. 2017.
[49] H. Cho, S.-J. Kang, and Y. H. Kim, "Image segmentation using linked mean-shift vectors and global/local attributes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 10, pp. 2132–2140, Oct. 2017.
[50] C. Panagiotakis, I. Grinias, and G. Tziritas, "Natural image segmentation based on tree equipartition, Bayesian flooding and region merging," *IEEE Trans Image Process*, vol. 20, no. 8, pp. 2276–2287, Aug. 2011.
[51] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.
[52] K. Zhang *et al.*, "A level set approach to image segmentation with intensity inhomogeneity," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 546–557, Feb. 2016.
[53] C. Du and S. Gao, "Image segmentation-based multi-focus image fusion through multi-scale convolutional neural network," *IEEE Access*, vol. 5, pp. 15750–15761, 2017.



ZHIKAI XING was born in Daqing, China, in 1993. He is currently pursuing the M.S. degree in control engineering with Northeast Forestry University, China. His research interests include image segmentation and intelligent optimization algorithm.

HEMING JIA received the Ph.D. degree in system engineering from Harbin Engineering University, China, in 2012. He is currently an Associate Professor with Northeast Forestry University. His research interests include nonlinear control theory and application, image segmentation, and swarm optimization algorithm.